



IPTC Standards



Specification Version 2.21
Power Conformance Level

Document Revision 1

International Press Telecommunications Council
Copyright © 2015. All Rights Reserved.
www.iptc.org



Copyrights and License

Copyright © 2015 by IPTC, the International Press Telecommunications Council. All Rights Reserved.

The IPTC NewsML-G2/EventsML-G2 specification document is published under the Creative Commons Attribution 4.0 license - see the full license agreement at <http://creativecommons.org/licenses/by/4.0/>. By obtaining, using and/or copying this document, you (the licensee) agree that you have read, understood, and will comply with the terms and conditions of the license.

This project intends to use materials that are either in the public domain or are available by the permission for their respective copyright holders. Permissions of copyright holder will be obtained prior to use of protected material. All materials of this IPTC standard covered by copyright shall be licensable at no charge.

Acknowledgments

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 (xport.net Ltd.), Dave Compton (Reuters), Jay Cousins (RivCom), Jean-Pierre Evain (EBU), John Evans (Transtel), Takahiro Fujiwara (EAST Co. Ltd.), Andreas Gebhard (Getty Images), Philipp Gortan (APA), Darko Gulija (HINA), Paul Harman (Press Association), Gerald Innerwinkler (APA), Johan Lindgren (Tidningarnas Telegrambyrå), Jayson Lorenzen (BusinessWire), Philippe Mouglin (AFP), Stuart Myles (initially 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	Date	Approved by	Remarks
2.0	2008-01-31	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
2.12	2012-06-13	IPTC Standards Committee	NewsML-G2 including EventsML-G2
2.15	2013-06-26	IPTC Standards Committee	NewsML-G2 including EventsML-G2
2.18	2014-06-18	IPTC Standards Committee	NewsML-G2 including EventsML-G2
2.21	2015-06-03	IPTC Standards Committee	NewsML-G2 including EventsML-G2

The specifications of NewsML-G2 and EventsML-G2 have been published separately up to the standard versions EventsML-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 under the main branding NewsML-G2 and in 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	2015-11-16	Michael Steidl	

About this Document

This document specifies the IPTC news exchange standard NewsML-G2 and its event focused 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 in subdirectories of <http://www.iptc.org/std/NewsML-G2/2.21/>

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 2.21 consists of (file names are added, # is to be substituted by the most current document revision number):

This Specification document - NewsML-G2_2.21-spec-PCL_#.pdf

XML Schema files applicable to the Core Conformance Level (see **Conformance Levels** on page 12):

- ◆ NewsML-G2_2.21-spec-All-Core_#.xsd

XML Schema files applicable to the Power Conformance Level (see **Conformance Levels** on page 12):

- ◆ NewsML-G2_2.21-spec-All-Power_#.xsd

All files above can be obtained from:

<http://www.iptc.org/std/NewsML-G2/2.21/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 a new version of the standard, this will be reflected by a new path to files and a new standard version number like NewsML-G2_2.21.
- ◆ 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 "3" in NewsML-G2_2.21-spec-Framework-Core_3.xsd.

The XML Schema files without the document revision number (e.g. "_3") 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.

Terminology

This document uses the terms MUST (NOT), SHOULD (NOT) and MAY as defined in [RFC2119].



Contents

1 Introduction to NewsML-G2	12
1.1 History	12
1.2 Conformance Levels	12
1.3 Supporting documents	12
2 Representing News - newsItem	13
2.1 Description	13
2.2 Indication of Compliance with a Standard and Conformance Level	13
2.3 Identification and Versioning	13
2.4 Catalog of Controlled Vocabularies	14
2.5 Signature Information	15
2.6 Rights Information	16
2.6.1 Processing Model	17
2.7 Item Metadata	22
2.7.1 Management Metadata	22
2.7.2 Processing the Publish Status of an Item	23
2.7.3 Processing of versionCreated	25
2.7.4 Best Practice for expressing an update or correction of an item	25
2.7.5 Best Practice for issuing a content warning	25
2.8 Item Links	26
2.8.1 Processing Links	26
2.9 News Content Metadata	27
2.9.1 Administrative Metadata	27
2.9.2 Descriptive Metadata	28
2.9.3 Other Content Metadata	29
2.10 Part of Content Metadata	29
2.10.1 Edit Units and Time Codes	29
2.10.2 Time Unit Types and Start/End Timestamp Formats	30
2.11 Assertions About Concepts	30
2.12 References to Inline Concepts	30
2.13 Document Derivation of Concepts	31
2.14 newsItem Content	31
2.15 News Content Characteristics	32
2.16 Channels	32
3 Introduction to EventsML-G2	33
3.1 Overview	33
3.1.1 What is EventsML-G2?	33
3.1.2 Business Advantages of Using EventsML-G2	33
3.1.3 What is an Event – to be represented by EventsML-G2	33
3.2 Definitions	34
3.2.1 Event Information	34
3.2.2 Coverage Information (LEGACY)	34
3.2.3 The Data Model	34
3.3 EventsML-G2 and iCalendar	34
4 Events	37
4.1 The Core Information about Events	37
4.2 Event Information in Items	38
4.2.1 An Event Concept in a Concept Item or Many Events in a Knowledge Item	38
4.2.2 Events in a NewsItem	39



5	Representing Concept Information - concept Component	41
5.1	Concept Component	41
5.2	Relationships Between Concepts	42
5.3	Details Associated with Specific Entities	43
5.3.1	Contact Information	43
5.3.2	Postal Address	43
6	Managing Individual Concepts - conceptItem	44
6.1	Description	44
6.2	Structure of a Concept Item	44
6.2.1	Note about the different identifiers for a concept and a conceptItem	44
6.3	Item Class	44
6.4	Concept related Metadata	44
6.5	Metadata Helpers	45
6.6	conceptItem Content	45
7	Managing Sets of Concepts - knowledgeItem	46
7.1	Description	46
7.2	Structure of a Knowledge Item	46
7.3	Item Class	46
7.4	Knowledge Related Metadata	46
7.5	Metadata Helpers	47
7.6	knowledgeItem Content	47
8	Packaging Items - packageItem	48
8.1	Structure of a Package Item	48
8.2	Item Class	48
8.3	Package Related Metadata	48
8.4	Metadata Helpers	48
8.5	packageItem Content	48
9	Planning news coverage - planningItem	50
9.1	Description	50
9.2	Structure of Planning Item	50
9.3	Item Class	50
9.4	Planning Related Metadata	50
9.5	Metadata Helpers	51
9.6	Planning Item Content	51
9.7	Processing Considerations	51
10	Dealing with Controlled Values	52
10.1	{scheme, code} Pair, Scheme URI and Concept URI	52
10.2	Qualified Code (QCode)	53
10.2.1	Lexical Space Specification and Processing Model for Scheme URIs, Scheme Aliases, Codes, and QCodes ..	53
10.3	Processing Catalogs	55
10.3.1	Structure of a Catalog	55
10.3.2	Processing Remote Catalogs	55
10.3.3	Caching a Catalog	55
10.3.4	Checking a Catalog	55
10.4	Processing Schemes	56
10.4.1	Evolution of Scheme URIs	56
10.4.2	Retrieving All Terms of a Scheme	56



- 10.5 Qualified and Typed Properties 56
- 10.6 Flexible Properties 57
- 10.7 Composite Concepts 58
- 10.8 Editing Attributes 58
- 11 Managing Catalogs - catalogItem 59
 - 11.1 Description 59
 - 11.2 Structure of a Catalog Item 59
 - 11.3 Item Class 59
 - 11.4 Catalog Related Metadata 59
 - 11.5 Catalog Item Content 59
- 12 Dealing with Labels and Blocks 60
 - 12.1 Introduction 60
 - 12.2 Internationalization Attributes 60
- 13 Exchanging Items - newsMessage 61
 - 13.1 Description 61
 - 13.2 Message Information 61
 - 13.3 About Using Schemes in a newsMessage 61
- 14 Specification Reference 63
 - 14.1 Introduction to the Common Components 63
 - 14.2 General Specifications 63
 - 14.2.1 XML Namespaces 63
 - 14.2.2 MIME Types 64
 - 14.2.3 Extension Points in XML 64
 - 14.2.4 Hint and Extension Points in XML 64
 - 14.3 Implementation Design Rules 64
 - 14.4 Processing Model Terminology 65
 - 14.5 Component Structure Format 65
 - 14.6 Element Definitions 67
 - 14.6.1 Access 67
 - 14.6.2 Access Status 68
 - 14.6.3 Accountable Person 69
 - 14.6.4 Action in Hop History 70
 - 14.6.5 Address Line 71
 - 14.6.6 Affiliation 72
 - 14.6.7 Alternative Identifier 73
 - 14.6.8 Alternative Locator 74
 - 14.6.9 Alternative Representation 75
 - 14.6.10 Anchor 76
 - 14.6.11 Assertion 77
 - 14.6.12 Assigned To 79
 - 14.6.13 Audience 80
 - 14.6.14 Bag 81
 - 14.6.15 Bag Item 82
 - 14.6.16 Broader 83
 - 14.6.17 By 84
 - 14.6.18 Capacity 85
 - 14.6.19 Catalog 86
 - 14.6.20 Catalog Container 87
 - 14.6.21 Catalog Item 88
 - 14.6.22 Channel of Remote Content 89
 - 14.6.23 Channel for News Message 92



14.6.24 Circle 93

14.6.25 Concept 94

14.6.26 Concept Definition 96

14.6.27 Concept Extension Property 97

14.6.28 Concept Identifier 98

14.6.29 Concept Item 99

14.6.30 Concept Name 100

14.6.31 Concept Note 101

14.6.32 Concept Reference 102

14.6.33 Concept Set 103

14.6.34 Confirmation 104

14.6.35 Contact Information 105

14.6.36 Content Metadata {Catalog Item} 106

14.6.37 Content Metadata {Concept Item} 107

14.6.38 Content Metadata {Knowledge Item} 109

14.6.39 Content Metadata {News Item} 111

14.6.40 Content Metadata {Package Item} 113

14.6.41 Content Metadata {Planning Item} 115

14.6.42 Content Metadata Extension Property 117

14.6.43 Content Provider 118

14.6.44 Content Set 119

14.6.45 Contributor 120

14.6.46 Copyright Holder 121

14.6.47 Copyright Notice 122

14.6.48 Country 123

14.6.49 Country Area 124

14.6.50 Creator 125

14.6.51 Credit Line 126

14.6.52 Date an Object Ceased to Exist 127

14.6.53 Date a Point Of Interest Ceased to Exist 128

14.6.54 Date Content Created 129

14.6.55 Date Content Modified 130

14.6.56 Date Item Embargo Ends 131

14.6.57 Date Item First Created 132

14.6.58 Date Item Version Created 133

14.6.59 Date of Birth of Person 134

14.6.60 Date of Creation of Object 135

14.6.61 Date of Creation of Point Of Interest 136

14.6.62 Date of Death of Person 137

14.6.63 Date of Dissolution of Geopolitical Area 138

14.6.64 Date of Dissolution of Organisation 139

14.6.65 Date of Foundation of Geopolitical Area 140

14.6.66 Date of Foundation of Organisation 141

14.6.67 Date of Transmission 142

14.6.68 Dateline 143

14.6.69 Dates 144

14.6.70 Recurrence Group 144

14.6.71 Date Resource Created 145

14.6.72 Description 146

14.6.73 Deliverable Of 147

14.6.74 Delivered Item Reference 148

14.6.75 Delivery Information 149

14.6.76 Derived From 150

14.6.77 Destination 151

14.6.78 Duration 152

14.6.79 Editorial Note 153

14.6.80 Editorial Service 154

14.6.81 Email Address 155



14.6.82 End Date/Time 156

14.6.83 Event 157

14.6.84 Event Details 158

14.6.85 Event Location 159

14.6.86 Events Wrapper 160

14.6.87 Excluded Audience 161

14.6.88 Exclusion Date 162

14.6.89 Exclusion Rule 163

14.6.90 Expiry Date 164

14.6.91 Facet (DEPRECATED) 165

14.6.92 Fax Number 166

14.6.93 File Name 167

14.6.94 G2 Content Type 168

14.6.95 G2 Item Class 169

14.6.96 Generator Tool 170

14.6.97 Genre 171

14.6.98 Geographic Position 172

14.6.99 Geopolitical Area Details 173

14.6.100 Group 174

14.6.101 Group Extension Property 176

14.6.102 Group Reference 177

14.6.103 Group Set 178

14.6.104 Hash Value 179

14.6.105 Has Financial Instrument 180

14.6.106 Headline 182

14.6.107 Hierarchy Info 183

14.6.108 Hop 184

14.6.109 Hop History 185

14.6.110 Icon 186

14.6.111 Incoming Feed Identifier 189

14.6.112 Inline Concept Marker 189

14.6.113 Inline Data 191

14.6.114 Inline Reference 194

14.6.115 Inline XML 195

14.6.116 Instance Of 198

14.6.117 Instant Messaging Address 199

14.6.118 Information Source 200

14.6.119 Item Class 201

14.6.120 Item Count 202

14.6.121 Item Metadata 203

14.6.122 Item Metadata Extension Property 205

14.6.123 Item Reference 206

14.6.124 Item Set 207

14.6.125 Item Title 208

14.6.126 Keyword 209

14.6.127 Knowledge Item 210

14.6.128 Language 211

14.6.129 Line 212

14.6.130 Line Break 213

14.6.131 Link 214

14.6.132 Locality 215

14.6.133 Located 216

14.6.134 Location 217

14.6.135 Location Details 218

14.6.136 Member Of 219

14.6.137 Message Header 220

14.6.138 Message Header Extension Property 221

14.6.139 Narrower 222



14.6.140 News Content Characteristics	223
14.6.141 News Coverage (Concept Item)	226
14.6.142 News Coverage (Planning Item)	228
14.6.143 News Coverage Extension Property	229
14.6.144 News Coverage Set	230
14.6.145 News Coverage Status	231
14.6.146 News Item	232
14.6.147 News Message	233
14.6.148 Object Details	234
14.6.149 Occurrence Status	235
14.6.150 Opening Hours	236
14.6.151 Origin	237
14.6.152 Organisation Details	238
14.6.153 Organiser	239
14.6.154 Original Representation	240
14.6.155 Package Item	241
14.6.156 Participant	242
14.6.157 Participation Requirement	243
14.6.158 Part of Content Metadata	244
14.6.159 Part of Content Metadata Extension Property	246
14.6.160 Party (Hop History)	247
14.6.161 Person Details	248
14.6.162 Phone Number	249
14.6.163 Planning Details	250
14.6.164 Planning Extension Property	251
14.6.165 Planning Item	252
14.6.166 POI Details	253
14.6.167 Polygon	254
14.6.168 Postal Address	255
14.6.169 Postal Address of a Point of Interest	256
14.6.170 Postal Code	257
14.6.171 Priority	258
14.6.172 Profile	259
14.6.173 Property Value Name	260
14.6.174 Publish Status	261
14.6.175 Rating	262
14.6.176 Recurrence Date	264
14.6.177 Recurrence Rule	265
14.6.178 Registration	266
14.6.179 Related Concept	267
14.6.180 Remote Catalog Reference	270
14.6.181 Remote Content	271
14.6.182 Remote Content Extension Property	274
14.6.183 Remote Information about a Concept	275
14.6.184 Rights Information	276
14.6.185 Rights Information Extension Property	277
14.6.186 Role in the Content Stream	278
14.6.187 Role in the Workflow	279
14.6.188 Ruby	280
14.6.189 Ruby Base	281
14.6.190 Ruby Parenthesis	282
14.6.191 Ruby Text	283
14.6.192 Same As	284
14.6.193 SameAs for a Scheme (DEPRECATED)	285
14.6.194 SameAs Scheme	286
14.6.195 Scheduled	287
14.6.196 Scheme Declaration	288
14.6.197 Scheme Metadata	289



14.6.198 Scheme Metadata Extension Property	290
14.6.199 Sender	291
14.6.200 Signal	292
14.6.201 Slugline	293
14.6.202 Span	294
14.6.203 Start Date/Time	295
14.6.204 Subject	296
14.6.205 Time Delimiter	297
14.6.206 Timestamp	299
14.6.207 Transmission Identifier	300
14.6.208 Type of a Concept	301
14.6.209 User Interaction	302
14.6.210 Urgency	303
14.6.211 Usage Terms	304
14.6.212 Visual Region Delimiter	305
14.6.213 Web Address	306
14.7 Element Group Definitions	307
14.7.1 Concept Definition Group	307
14.7.2 Concept Relationships Group	307
14.7.3 Entity Details Group	307
14.7.4 Administrative Metadata Group	308
14.7.5 Knowledge Descriptive Metadata Group	308
14.7.6 Descriptive Metadata Core Group	308
14.7.7 Descriptive Metadata Group	309
14.7.8 Item Management Group	309
14.8 Datatype Definitions	310
14.8.1 Any Item Type	310
14.8.2 Approximate Date and Time Property Type	311
14.8.3 Audience Type	312
14.8.4 Block Type	313
14.8.5 Concept Identifier Type	315
14.8.6 Content Metadata Type	316
14.8.7 Date and Optional Time Property Type	317
14.8.8 Date and Optional Time Type	318
14.8.9 Date and Time or Null Value Property Type	319
14.8.10 Date and Time Property Type	320
14.8.11 Electronic Address Type	321
14.8.12 Electronic Address Tech Type	322
14.8.13 Flexible 1 Concept Property Type	323
14.8.14 Flexible 1 Extension Property Type	324
14.8.15 Flexible 1 Party Property Type	324
14.8.16 Flexible 1 Property Type	326
14.8.17 Flexible 2 Extension Property Type	328
14.8.18 Flexible Location Property Type	328
14.8.19 Flexible Organisation Property Type	330
14.8.20 Flexible Party Property Type	332
14.8.21 Flex Person Property Type	334
14.8.22 Flexible Property Type	336
14.8.23 Flexible Property 2 Type	338
14.8.24 Integer 0 to 100 Type	340
14.8.25 Integer 1 to 9 Type	341
14.8.26 International String Type	342
14.8.27 IRI List Type	343
14.8.28 IRI Type	344
14.8.29 Label 1 Type	345
14.8.30 Link 1 Type	346
14.8.31 Normalized String Type	349



- 14.8.32 QCode List Type 349
- 14.8.33 QCode Property Type 350
- 14.8.34 QCode Type 351
- 14.8.35 Qualified Property Type 352
- 14.8.36 Recurrence Rule Type 353
- 14.8.37 Related Concept Type 354
- 14.8.38 Rights Label Type 355
- 14.8.39 Truncated Date and Time Property Type 356
- 14.8.40 Truncated Date and Time Type 357
- 14.8.41 Typed Qualified Property Type 358
- 14.8.42 Versioned String Type 359
- 14.9 Attribute Group Definitions 360
 - 14.9.1 Internationalization Attributes Group 360
 - 14.9.2 Common Power Attributes Group 360
 - 14.9.3 Time Validity Attributes Group 361
 - 14.9.4 Ranking Attributes Group 362
 - 14.9.5 Qualify Attributes Group 363
 - 14.9.6 Quantify Attributes Group 363
 - 14.9.7 News Content Attributes 364
 - 14.9.8 Target Resource Attributes Group 365
 - 14.9.9 News Content Characteristics 365
- 15 Glossary 369
- 16 References 372
- 17 Index 373
 - 17.1 Elements 373
 - 17.2 Datatypes 376



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 IPTC, and it was decided to create a consistent set of complementary standards as a comprehensive and inter-operable 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, it includes NewsML-G2, EventsML-G2 and SportsML-G2; NewsML-G2 is the brand name for all of them.

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 are common across the members of the G2-Standards.

To better understand the terminology used in the G2-Standards specifications we recommend the **Glossary** (page 369) as a reference, as it provides an extensive set of terms and their definitions.

Since the initial release of NewsML-G2 in 2008 many news providers have adopted this standard and IPTC has extended and slightly modified the specifications by raised change requests.

To reflect implementations IPTC has conducted a survey of the properties used in practice in 2013 and the resulting “mainstream profile” is shown in chapter 19 of the G2 Implementation Guide, see below in 1.3.

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 Conformance Level.

1.3 Supporting documents

This document and the XML Schema files are the formal specification of the G2 Standards.

IPTC also provides documents supporting the implementation of the standards in subfolders of <http://www.iptc.org/std/NewsML-G2/2.21/>

- ◆ G2 Quick Start Guides: small documents explaining how to take the first steps for successfully start with text, photos, video and packages thereof using NewsML-G2
- ◆ The G2 Implementation Guide: a comprehensive guideline covering also special work areas like the management of controlled vocabularies and migrating from existing standards to NewsML-G2.
- ◆ A set of almost 30 NewsML-G2 and EventsML-G2 example XML document.
- ◆ A Structure Matrix table showing for each property its attributes.



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 232) *standard* attribute **MUST** be set to “NewsML-G2” from NewsML-G2 2.9 on. “EventsML-G2” **MAY** be used up to version 1.7 of this standard.

The schema version to which the newsItem conforms **MUST** be indicated as an attribute. The current version is identified by the string “2.21”.

The IPTC conformance level to which the newsItem conforms in this specification **MUST** be indicated by the conformance attribute value “power”.

Sample Core Conformance Level:

```
<newsItem standard="NewsML-G2" standardversion="2.21"
  xmlns="http://iptc.org/std/nar/2006-10-01/" >
</newsItem>
```

Sample Power Conformance Level:

```
<newsItem standard="NewsML-G2" standardversion="2.21" conformance="power"
  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 holding a persistent and globally unique identifier. IPTC recommends using an IRI but this is not a requirement. Any string capable of acting as a globally unique identifier may be used.



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="2.21" conformance="power"
  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 86) is defined as a set of scheme declarations in use by a news provider for a given Item.

Due to the large number of the same schemes potentially used in many single Items, and knowing that bandwidth is very important to the News industry, the catalog may be stored remotely and referenced by the Item (see **catalogRef** on page 270).

A remote catalog *MUST* have a *href* attribute which contains the URL of a remote catalog. A remote catalog takes the form of an XML file with a catalog element as root. (An XML requirement is to add the NewsML-G2 namespace definition to the catalog element.)

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

- ◆ The URL of a remote catalog *MUST NOT* be relative.
- ◆ If a remote catalog is functionally changed, the URL used to access it *MUST* be changed. Functional changes are:
 - the addition or removal of a scheme declaration,
 - a change to any of the scheme aliases,
 - a change to any of the scheme URIs.
 - a change to any of the combinations of schema alias and scheme URI.

One or more additional titles (see **title {itemMeta}** on page 208) for a catalog or catalogRef *MAY* be provided in different languages and variants.

To extend the information about the catalog some optional attributes of the catalog element may be used:

- url: defines the location of the catalog as remote resource.
- authority: defines the authority controlling this catalog
- guid: a Globally Unique Identifier for this kind of catalog as managed by a provider
- version: version corresponding to the guid of the catalog

As some required properties of any G2 Item 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 **sameAsScheme** (page 286) 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 to a concept if the scheme identifier of this concept (used as property value) matches the scheme identifier in the sameAsScheme child in the catalog.

Another reason for establishing a sameAsSchema 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 52).

Sample:

```
<newsItem standard="NewsML-G2" standardversion="2.15"
  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 276) 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 69) person, a **copyrightHolder** (page 121) and a set of **copyrightNotice** (page 122) elements and **usageTerms** (page 304).

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 63) and **Extension Points in XML** (page 64).

2.6.1 Processing Model

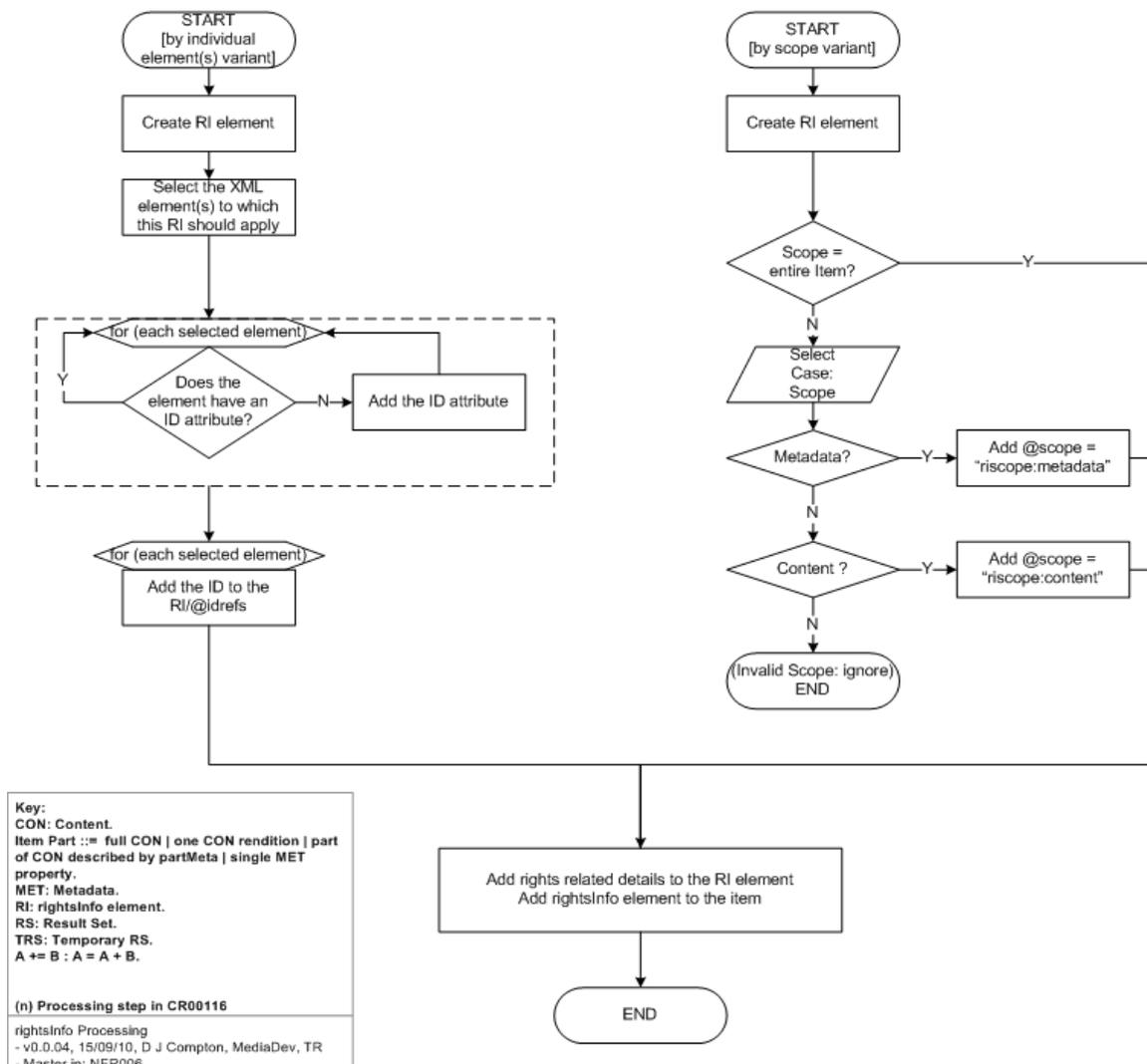
Rules for adding rightsInfo expressions - taking the News Provider View:

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

Use Case 1 / 3:

Rules for adding <rightsInfo> expressions – taking the News Provider View:

To be answered: How to apply rightsInfo elements referencing only a fraction of a G2 Item?



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.

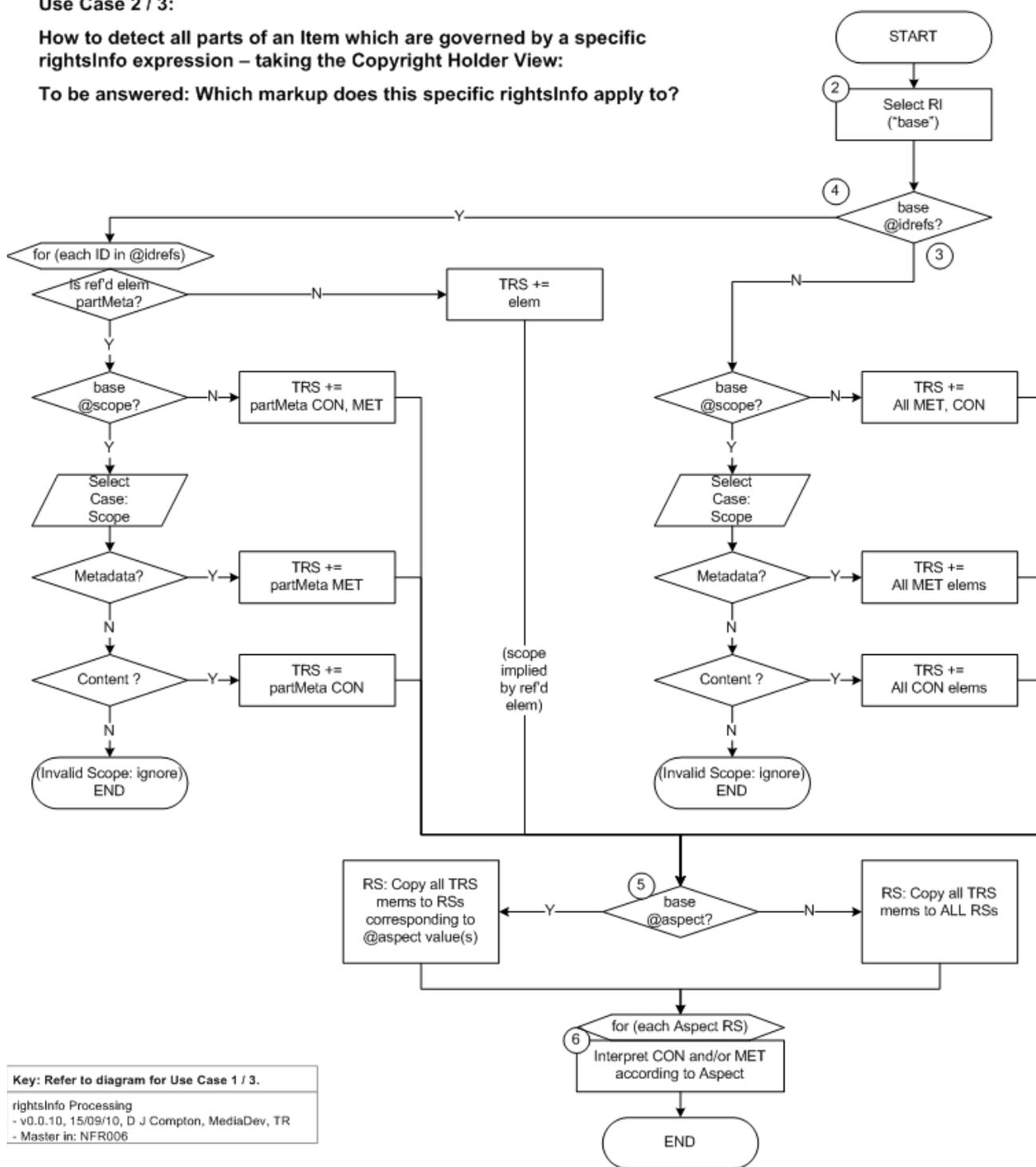
How to detect all parts of an Item which are governed by a specific rightsInfo expression - taking the Copyright Holder View:

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

Use Case 2 / 3:

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?



Key: Refer to diagram for Use Case 1 / 3.
rightsInfo Processing
- v0.0.10, 15/09/10, D J Compton, MediaDev, TR
- Master in: NFR006

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:

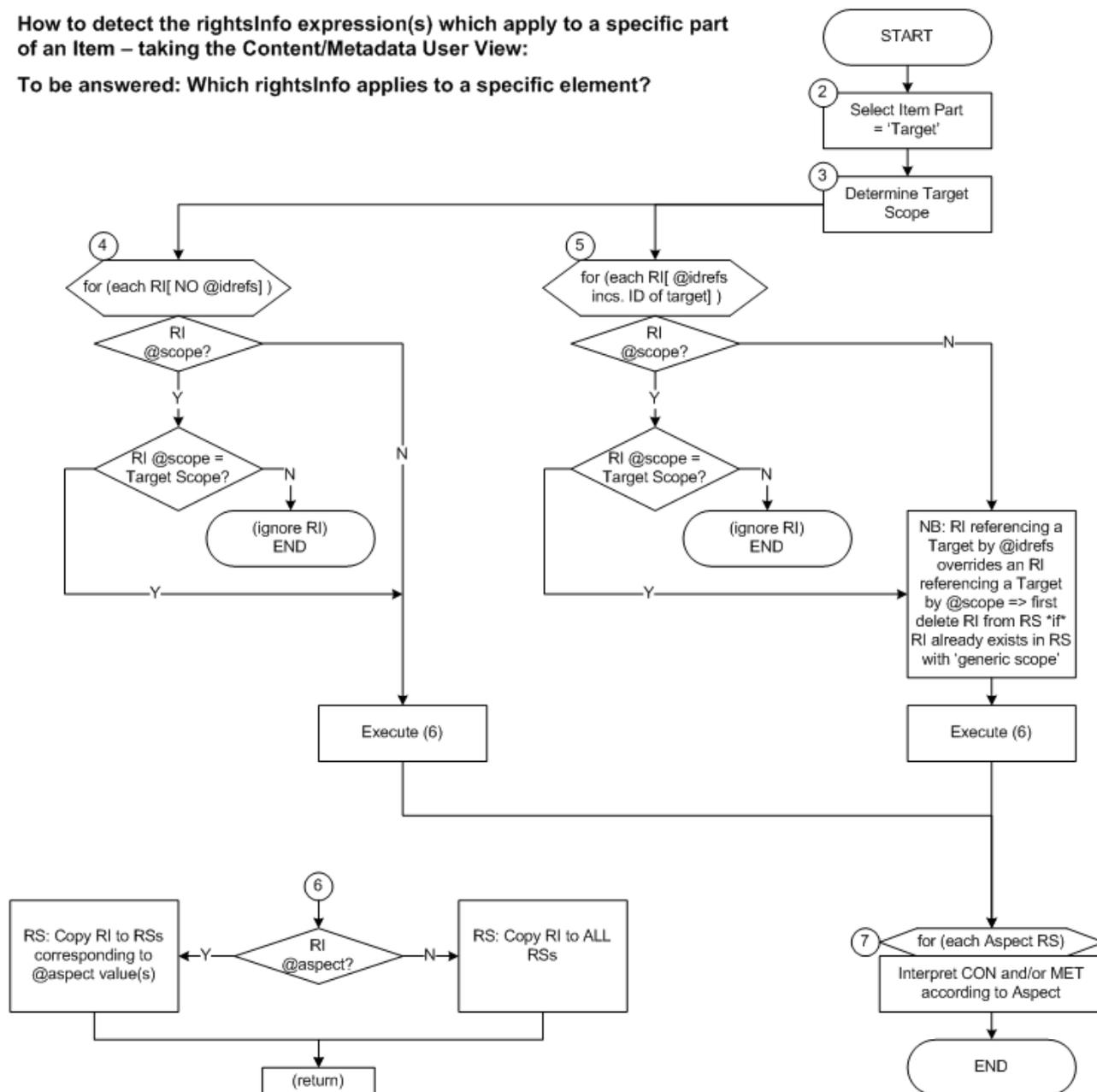
To be answered: Which rightsInfo applies to a specific element?



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.
RightsInfo 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 mandatory `itemMeta` (page 203) 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)	201
Content Provider	provider	(1)	118
Date Item Version Created	versionCreated	(1)	133
Date Item First Created	firstCreated	(0..1)	132
Date Item Embargo Ends	embargoed	(0..1)	131
Publish Status	pubStatus	(0..1)	261
Role in the Workflow	role	(0..1)	279
File Name	filename	(0..1)	167
Generator Tool	generator	(0..1)	170
Profile	profile	(0..1)	259
Editorial Service	service	(0..unbounded)	154
Item Title	title {itemMeta}	(0..unbounded)	208
Editorial Note	edNote	(0..unbounded)	153
Member Of	memberOf	(0..unbounded)	219
Instance Of	instanceOf	(0..unbounded)	198
Signal	signal	(0..unbounded)	292
Alternative Representation	altRep	(0..unbounded)	75
Deliverable Of	deliverableOf	(0..1)	147
Hash Value	hash	(0..unbounded)	147

The IPTC provides a mandatory standardised scheme applicable to the **itemClass** (page 201) 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 63) and **Extension Points in XML** (page 64).

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.

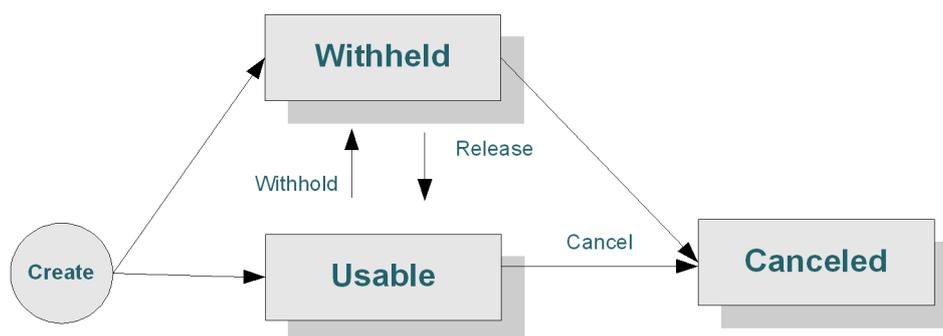
Embargoes are managed by the **embargoed** (page 131) property. At the level of the data model the embargoed element could be linked now to an **edNote** (page 153) element if the existing embargoed is empty (<embargoed />).

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 261) 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 recipient 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 261) and embargoed properties:

```
Test pubStatus = canceled:
    The Item must not be used, ever. Any usage of the Item must be prohibited, 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 133) 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 initial 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 initial 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 or 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 292) 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 161) 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 214) 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 361) (*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 214) 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 75)) 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 which applies to the whole content of the Item is wrapped in the contentMeta wrapper element and split between administrative and descriptive metadata. Be aware that some G2 Items adopt only a subset of the metadata properties listed below. Information about a part of the content is wrapped by **Part of Content Metadata** (page 29).

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)	303
Date Content Created	contentCreated	(0..1)	129
Date Content Modified	contentModified	(0..1)	130
Located	located	(0..unbounded)	216
Information Source	infoSource	(0..unbounded)	200
Creator	creator	(0..unbounded)	125
Contributor	contributor	(0..unbounded)	120
Audience	audience	(0..unbounded)	80
Excluded Audience	exclAudience	(0..unbounded)	161
Alternative Identifier	altId	(0..unbounded)	73

2.9.1.1 Dates Processing Model

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

contentCreated (page 129) and **contentModified** (page 130) 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 133).

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 80)) and a set of “negative” values (**exclAudience** (page 161)). 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)	211
Genre	genre	(0..unbounded)	171
Keyword	keyword	(0..unbounded)	209
Subject	subject	(0..unbounded)	296
Slugline	slugline	(0..unbounded)	293
Headline	headline	(0..unbounded)	182
Dateline	dateline	(0..unbounded)	143
By	by	(0..unbounded)	84
CreditLine	creditline	(0..unbounded)	126
Description	description	(0..unbounded)	146



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 63) and [Extension Points in XML](#) (page 64).

2.10 Part of Content 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 244) 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 186) - 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 297) element. The time scope is expressed as *start* and *end* timestamp attributes plus an additional time unit (*timeunit*) attribute. Both timestamp values MUST be within the overall content duration.

A region of an image MAY be defined by a [regionDelim](#) (page 305) 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 278) 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 63) and [Extension Points in XML](#) (page 64).

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.

$\text{EditUnit} = 1 / \text{EditRate}$.

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.

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 297).

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).

2.11 Assertions About Concepts

When a concept is used as the value of many properties or by a property with a limited granularity of concept details, it may be useful to group supplemental information about this concept at a unique location.

The optional and repeatable **assert** (page 77) property provides information about a concept identified by a qualified code, a full URI or a literal value. 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. Any changes applied to a concept after that time are not reflected by an assert element.

2.12 References to Inline Concepts

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

The optional and repeatable **inlineRef** (page 194) 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 or a full URI, 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.



2.13 Document Derivation of Concepts

An increasing number of metadata values is not added explicitly by human interaction but by an automated derivation using some kind of knowledge network. In this case it could be of value to indicate the concept(s) from which a specific value of a metadata property has been derived. For this purpose the optional **derivedFrom** (page 150) element can be used.

The `qcode` or `uri` attribute of this element define the concept from which another concept has been derived. The `idref` attribute of this element refers to the `id` attributes of all the properties in this G2 item having a value which has been derived from the concept represented by the `derivedFrom` property.

2.14 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 119) 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 be 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 195) 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 191) 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 271) wrapper element may be used for representing any kind of media and data format. The data is stored independently of the `newsItem` and is referenced via a hyperlink (`href`). The size in bytes of the remote content MAY be indicated. The element MAY also have **time-ValidityAttributes** (page 361) (*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 this case alternative locations of the content are provided by `altLoc` child elements of one `remoteContent`, multiple `remoteContent` elements with the same rendition value SHOULD NOT be used.

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 rea-



son 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.

2.15 News Content Characteristics

newsContentCharacteristics (page 365) 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.

2.16 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 {News Item}** (page 89) element MAY therefore be defined as a child of a **remote-Content** (page 271) 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 specification extends the NewsML-G2 structural specification with some event-specific details and adds well defined functionality for conveying events.

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 a 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 is:

- ◆ 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 structures may be added to capture further 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 NewsML-G2 Architecture allowing for the reuse of 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.

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 data exchange. 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.

Note: information about the planned coverage of an event can be shared by using a Planning Item - see [Planning news coverage - planningItem](#) (page 50)

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 newer and more powerful tool for expressing and managing the planned coverage of events: [Planning news coverage - planningItem](#) (page 50). To provide backward compatibility the structure for coverage information as part of an event structure is still valid, but it is strongly recommended to separate out the planning information into the Planning Item, enabling event definition and planning to be decoupled.

The old-style coverage information describes the plan of news coverage for this event but it is highly recommended to adopt the new-style Planning Item.

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:

- ◆ Persistent 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 NewsML-G2-Concept Item which is a generic NAR structure for concepts extended by a set of detailed information specific to an event. As for any other kind of Concept 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 Concept in a Concept Item or Many Events in a Knowledge Item](#) (page 38).

- ◆ 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 39).

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 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)
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



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

iCalendar Event Component (Alphabetically)	Corresponding EventsML-G2 Component
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 of whether the information about an event is topical or persistent (see [The Data Model](#) on page 34) the same structure is used to mark it up.

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

- ◆ A natural language **name** (page 100) for the event. This name should be rather concise and can be expressed in different languages.
- ◆ A natural language **definition** (page 96) for the event which can be more extensive than the name; it can explain facets in detail.

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 101) 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 284), **broader** (page 83), **narrower** (page 222) and **related** (page 267) can be used to define a relationship between this event and 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.

Additionally, a set of rather event-specific properties - wrapped by the **eventDetails** (page 158) property:

- ◆ A **dates** (page 144) sub-structure expresses the start date and the end date or duration of the event. This includes using the "approximative dates", i.e. a range of dates, 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).

- ◆ **occurStatus** (page 235) indicates the status of the occurrence - if this is an unplanned or planned event, and if it is planned - how likely it is to occur.
- ◆ A set of **registration** (page 266) information which defines how persons have to register for the event, this may include the accreditation of journalists.
- ◆ A set of **accessStatus** (page 68) information.
- ◆ A set of **participationRequirement** (page 243) properties. This can 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 296) properties expressing 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 217) properties. In most cases it will be the single location where the event will take place - but e.g. festivals could have more than one location.
- ◆ A set of **participant** (page 242) 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 239) 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 105) 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 211) properties reflecting all “official” languages at the event.
- ◆ A **newsCoverage {Concept}** (page 226) 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 **Planning news coverage - planningItem** (page 50).
- ◆ 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 An Event Concept in a Concept Item or Many Events in a Knowledge Item

The persisting knowledge facet of event information is represented as a Concept (see **Representing Concept Information - concept Component** (page 41))

As all other concepts a single event can be managed by a Concept Item (see **Managing Individual Concepts - conceptItem** on page 44), and subsequently many of them by a Knowledge Item (see **Managing Sets of Concepts - knowledgeItem** on page 46).

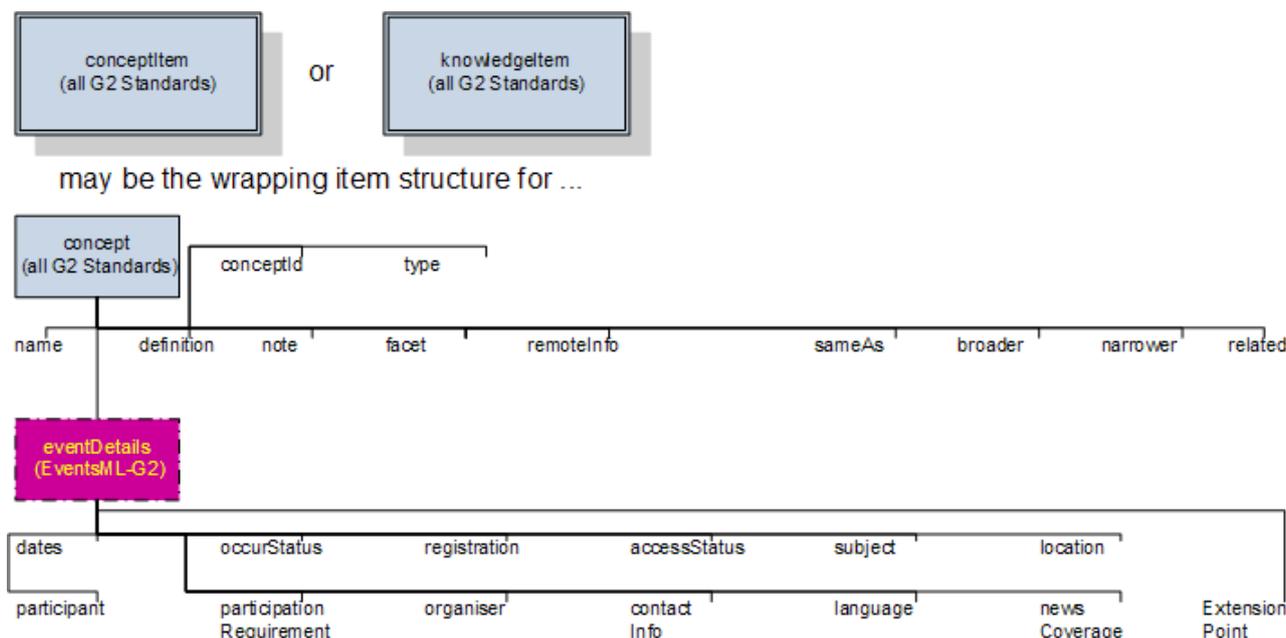
Any Concept Item or Knowledge Item provides a group of generic definitions and a set of details specific to a kind of concept, in this case specific to an event.

Event concepts use the generic part of a concept in order to define:

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

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 can be used by other items (e.g. News Items or Concept Items) to reference this event. By a purely technical view this Concept Identifier can be used as value of any property referring to a concept. On a semantic level it is required that the semantics of this property includes events as allowed concepts – e.g. a property which is 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 text, photo, audio or video covering the event.
- ◆ Using an event's Concept Identifier with the **Same As** (page 284), **Broader** (page 83), **Narrower** (page 222) and **Related Concept** (page 267) 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.

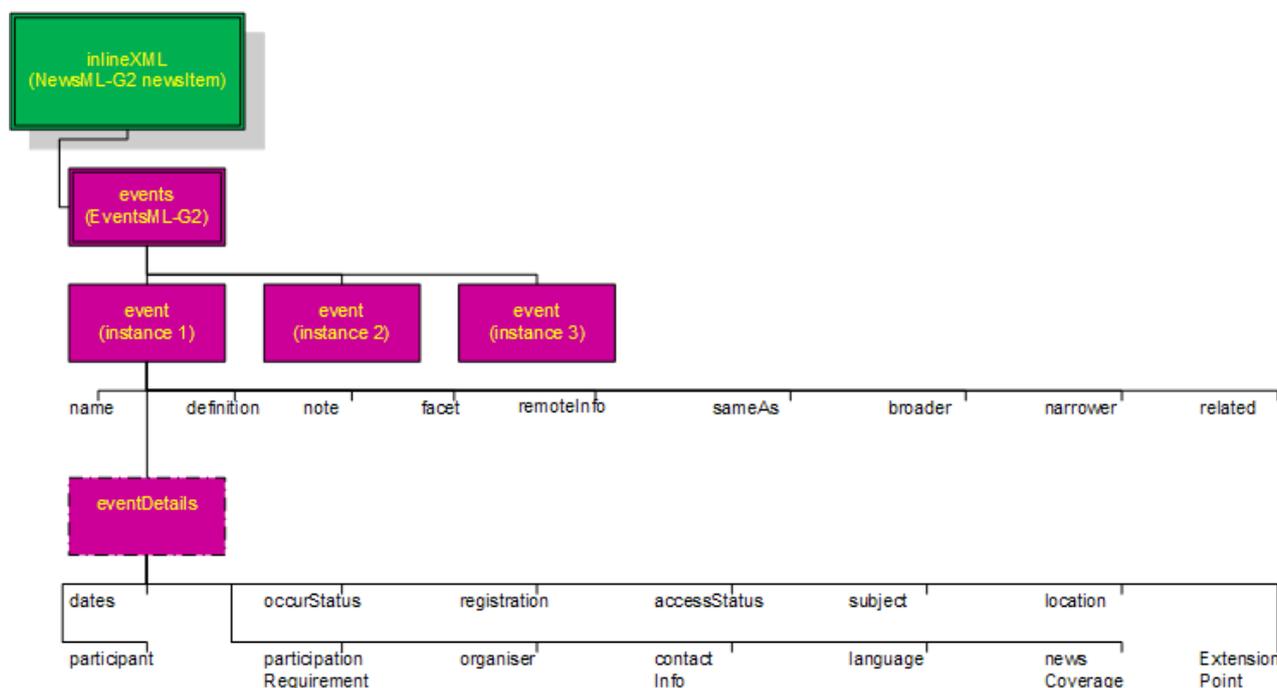
A provider could think of this use case scenario: a "top events of the next weekend" Knowledge Item is circulated with event concepts on Monday. On Wednesday, a new version of this Knowledge Item is sent with updated events, and cancelled events removed.

4.2.2 Events in a NewsItem

Topical event information may be conveyed by using the NewsML-G2 NewsItem (see **Representing News - newsitem** on page 13) as a wrapping item instance. The structure of a NewsItem defines a special node where content plug-ins can be attached: the inlineXML element.

For EventsML-G2 an **Events Wrapper** (page 160) element has to be added as a child to **Inline XML** (page 195) and it acts as a wrapper of one-to-many **Event** (page 157) 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 generic 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.

4.2.2.1 News Metadata

In general the News Metadata section of a NewsItem - wrapped by the **Content Metadata {News Item}** (page 111) 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 metadata - 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. politics, 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 94) element provides a set of properties shared by all types of concept.

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 may be identified by different identifiers from different company vocabularies. 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.

The properties common to all types of concepts are:

A concept **MUST** have a concept identifier, expressed as a **conceptId** (page 98) 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 301) 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 100) and **MAY** be further defined in natural-language by a **definition** (page 96) or **note** (page 101) and by **remoteInfo** (page 275). Definition and note are repeatable and **MAY** be specified in multiple languages.

Different variants of a name are allowed. The *role* attribute refines the semantics 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 definition, note and remoteInfo **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 183) 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 284) child element.

The sameAs element **MUST** have either a *qcode* or a *uri* or a *literal* attribute which identifies a concept, for the exact rules see the table below in the chapter **Relationships Between Concepts** (page 42). 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 326). *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 267) 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 83), and **narrower** (page 222) and further MAY add a more flexible **related** (page 267) 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 is that all of them or none of them MUST NOT be used in any case. The following table defines how the attributes MUST be used with the different properties, when establishing a relationship. (Be aware that establishing a relationship to an arbitrary value is specific to the **related** (page 267) property only, the specification section of this element provides more details.)

Table 7. Which attributes to use with relationship properties

Property	Attribute qcode or uri	Attribute literal	Set of attributes of an arbitrary value	Use case
<i>sameAs</i>	Yes	No	No	1
<i>narrower</i>	Yes	No	No	1
	No	Yes	No	2
<i>broader</i>	Yes	No	No	1
	No	Yes	No	2
<i>related</i>	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 326)).

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 58).



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 248) 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 238) 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; if this address is used only for a formal registration and the organisations business office does not reside there it should not be used for making direct contact with this company.

geoAreaDetails (page 173) 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 the coordinate reference system (*gpsdatum* attribute) expressed as a string. In the absence of this attribute, the WGS84 system is assumed.

POIDetails (page 253) 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 234) include a date of creation, a creator of the object and a copyright notice.

5.3.1 Contact Information

contactInfo (page 105) 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 it possible to group together all information belonging of the same nature.

Contact information includes email addresses, instant messaging addresses (im), international phone numbers, international fax numbers, web addresses, postal addresses and notes. 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 255)) 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 99) aims to convey knowledge about a single concept, a named entity such as an organisation or an abstract notion such as a news subject (see [Representing Concept Information - concept Component](#) (page 41)). Typically a conceptItem holds only a rather limited set of metadata about the concept and the structured concept data as content of the item.

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 Concept Items, managed by different providers, may contain structured information about the same concept.

6.2 Structure of a Concept Item

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.

6.2.1 Note about the different identifiers for a concept and a conceptItem

Each concept has its globally unique concept identifier (see [conceptId](#) (page 98)) which is part of the concept structure and defined by the authority of the scheme.

Each conceptItem has its globally unique identifier (guid) attribute which is assigned by a system managing G2 items.

Be aware that these two identifiers must not be mixed up, all references to a concept MUST use the concept identifier and not the guid of the conceptItem.

6.3 Item Class

The IPTC provides a mandatory standardised scheme applicable to the **itemClass** (page 201) 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.



The set of descriptive metadata for a Concept Item 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)	211
Keyword	keyword	(0..unbounded)	209
Subject	subject	(0..unbounded)	296
Slugline	slugline	(0..unbounded)	293
Headline	headline	(0..unbounded)	182
Description	description	(0..unbounded)	146

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 63) and **Extension Points in XML** (page 64).

Please review **News Content Metadata** (page 27) of the News Item chapter for more information.

6.5 Metadata Helpers

The conceptItem includes three properties which are available to help make metadata assertions:

- ◆ the assert property: helps to bundle and extend details of concepts, see **Assertions About Concepts** (page 30)
- ◆ the inlineRef property: helps to reference concepts which are inline, within free text properties of type label, see **References to Inline Concepts** (page 30)
- ◆ the derivedFrom property: helps to document from which concept a concept, used as property value in this item, has been derived, see **Document Derivation of Concepts** (page 31)

6.6 conceptItem Content

The content of a conceptItem is a concept component (see **Concept Component** on page 41).

7 Managing Sets of Concepts - knowledgeltem

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 **knowledgeltem** (page 210) bundles a set of concept components which are managed and exchanged as a whole. A knowledgeltem 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 knowledgeltem 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 knowledgeltems are the taxonomy of IPTC Subject NewsCodes or an authority list of people’s descriptions maintained by a given provider. Typical characteristics of a knowledgeltem are:

- ◆ It contains a set of concepts components covering a specific purpose, e.g. concepts from a single scheme, concept from different schemes and relevant in the context of a specific topic.
- ◆ 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 Knowledge Item

The model of a knowledgeltem 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 13) for more information.

7.3 Item Class

The IPTC provides a mandatory standardised scheme applicable to the **itemClass** (page 201) property, identified by the URI <http://cv.iptc.org/newscodes/cinature/>.

7.4 Knowledge Related Metadata

Metadata about the whole set of concepts held by a Knowledge Item are wrapped by the contentMeta element.

Metadata about specific concepts held by a Knowledge Item are wrapped by one to many partMeta elements. A typical use case of partMeta is to indicate when several concepts were modified at the same time, by associating those concepts with a specific partMeta which has the associated contentModified property.

The set of administrative metadata is common to all classes of Items.

The set of descriptive metadata for a Knowledge Item 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)	211
Keyword	keyword	(0..unbounded)	209
Subject	subject	(0..unbounded)	296
Slugline	slugline	(0..unbounded)	293
Headline	headline	(0..unbounded)	182
Description	description	(0..unbounded)	146



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 63) and [Extension Points in XML](#) (page 64).

Please review [News Content Metadata](#) (page 27) of the News Item chapter for more information.

7.5 Metadata Helpers

The `knowledgItem` includes three properties which are available to help make metadata assertions:

- ◆ the `assert` property: helps to bundle and extend details of concepts, see [Assertions About Concepts](#) (page 30)
- ◆ the `inlineRef` property: helps to reference concepts which are inline, within free text properties of type label, see [References to Inline Concepts](#) (page 30)
- ◆ the `derivedFrom` property: helps to document from which concept a concept, used as property value in this item, has been derived, see [Document Derivation of Concepts](#) (page 31)

7.6 knowledgItem Content

A `conceptSet` (page 103) wrapper element contains a set of concept components (see [Concept Component](#) on page 41). 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 `knowledgItem`.



8 Packaging Items - `packagelitem`

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

A `packagelitem` (page 241) facilitates the packaging of all kinds of Items, from really simple constructs to the highly hierarchical structures created by some news providers.

Examples of `packagelitem`s are a collection of pictures, a “top ten” list of `newsItem`s, an unordered set of `newsItem`s 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 `packagelitem` 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 Package Item

The model of a `packagelitem` is very similar to the model of a `newsItem` (page 232). 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.

8.2 Item Class

The IPTC provides mandatory standardised schemes applicable to the `itemClass` property of a `packagelitem`, 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 `packagelitem`s and `newsItem`s. Please review [News Content Metadata](#) (page 27) of the News Item chapter for more information.

8.4 Metadata Helpers

The `packagelitem` includes three properties which are available to help make metadata assertions:

- ◆ the `assert` property: helps to bundle and extend details of concepts, see [Assertions About Concepts](#) (page 30)
- ◆ the `inlineRef` property: helps to reference concepts which are inline, within free text properties of type label, see [References to Inline Concepts](#) (page 30)
- ◆ the `derivedFrom` property: helps to document from which concept a concept, used as property value in this item, has been derived, see [Document Derivation of Concepts](#) (page 31)

8.5 `packagelitem` Content

A `groupSet` (page 178) represents a tree of components, a component can be

- ◆ a `group` (page 174) element which contains one to many of the components below
- ◆ an `itemRef` (page 206) element referring to a package-external G2 item or a web resource
- ◆ a `groupRef` (page 177) element referring to another group of this Package Item

All G2 items included into a package are included by reference, as physical inclusion would break the capability to manage inner Items independently of the outer Package Item.

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

There **MUST** be at least one group element in the `groupSet` but there could also be many of them. In any case the value of `root` attribute of the `groupSet` element **MUST** be the `id` attribute value of the group acting as a root.



A group component may contain references to other group components (using a groupRef element with its *idref* attribute) of the same package item and/or references to Items or Web resource (using the itemRef element 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 implementing the mode "bag".

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.

Further there are other attributes available, see the specification table of the element.

Sample:

```
<groupSet root="g1">
  <group id="g1" mode="mode:seq" role="grouprole:main">
    <groupRef idref="g2"/>
    <itemRef guidref="urn:newsml:iptc.org:20070530:tutorial-text-xhtml"/>
  </group>
  <group id="g2" role="grouprole: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 252) 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 Planning Item

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 13) for more information.

9.3 Item Class

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

9.4 Planning Related Metadata

The set of administrative metadata is common to all classes of Items.

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

Table 10. Descriptive Metadata Core Group Elements

Element Title	Element Name	Card	Described on Page
Language	language	(0..unbounded)	211
Keyword	keyword	(0..unbounded)	209
Subject	subject	(0..unbounded)	296
Slugline	slugline	(0..unbounded)	293
Headline	headline	(0..unbounded)	182
Description	description	(0..unbounded)	146

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 63) and [Extension Points in XML](#) (page 64).



Please review [News Content Metadata](#) (page 27) of the News Item chapter for more information.

9.5 Metadata Helpers

The `planningItem` includes three properties which are available to help make metadata assertions:

- ◆ the `assert` property: helps to bundle and extend details of concepts, see [Assertions About Concepts](#) (page 30)
- ◆ the `inlineRef` property: helps to reference concepts which are inline, within free text properties of type label, see [References to Inline Concepts](#) (page 30)
- ◆ the `derivedFrom` property: helps to document from which concept a concept, used as property value in this item, has been derived, see [Document Derivation of Concepts](#) (page 31)

9.6 Planning Item Content

A `newsCoverageSet` (page 230) 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 use three `newsCoverage` components.

The `newsCoverage {Planning}` (page 228) component holds the mandatory planning property and the optional delivery property.

The `planning` (page 250) 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 149) 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 147) 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 or knowledge management company. 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 53) 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/mediatopic/>". If the code "15000000" represents the concept of "Sport", then the concept URI for "Sport" would be "<http://cv.iptc.org/newscodes/mediatopic/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 14) 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 web 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 web 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 201) or **pubStatus** (page 261).

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 351) 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** (page 288) 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 (#u003A) and white space (#u0020 | #u0009 | #u000D | #u000A).
- ◆ Lexical space for Codes:
 - All 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. See also section 10.2.1.2 (Creating Codes) below.
 - As an alternative to percent-encoding whitespace characters (#u0020 | #u0009 | #u000D | #u000A) as defined by RFC 3986, these characters may be replaced by a sequence of one or more unreserved characters - like e.g. underscore or hyphen - 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 in that scheme

10.2.1.2 Processing Model

10.2.1.2.1 Creating Scheme URIs

Define a URI complying with the rules defined in 10.1 and 10.2.1.

Note: every scheme URI must comply with the RFC defining URIs (3986) or IRIs (3987).

10.2.1.2.2 Creating Codes

As defined in section 10.1 a concept URI is created by appending the code of a concept to the scheme URI of the vocabulary.

Therefore, appending a Code to a valid Scheme URI must make a valid URI, in particular, the Code must only contain characters that are legal URI characters (RFC 3986). As defined in RFC 3986, the Scheme Authority MAY percent encode reserved characters as well as the percent ("%") character depending on the role of each character as defined by that specific publisher for this Concept URI. Any percent encoding which is applied to characters in the code of a Concept URI MUST be used also by the corresponding QCode.

The Scheme Authority is fully responsible that concept URIs defined by him are properly resolved by any resolution system provided by him.

The examples below show how to deal with a string which should be used as code and includes a reserved character:

Example **without** encoding a slash in the code:

String to be used as Code: ebc13/14

Code: ebc13/14

Scheme URI: <http://cv.example.org/schemeA/>

Scheme Alias: schA

QCode: schA:ebc13/14

Concept URI: <http://cv.example.org/schemeA/ebc13/14>

Note: The Scheme Authority has to take care that this URL is resolved properly by a resolution system provided by him.

Example **with** encoding a slash in the code:

String to be used as Code: ebc13/14

Code: ebc13%2F14 (with applied percent encoding)

Scheme URI: <http://cv.example.org/schemeB/>

Scheme Alias: schB

QCode: schB:ebc13%2F14

Concept URI: <http://cv.example.org/schemeB/ebc13%2F14>

10.2.1.2.3 QCodes

10.2.1.2.3.1 Creating QCodes

Concatenate the Scheme Alias, a colon and the Code to form a QCode.

10.2.1.2.3.2 Inserting a QCode as the value of an attribute in a G2 XML document

1. Take a QCode as created in 10.2.1.2.3.1 and apply any required XML encoding to this string (Note: this is typically done by the XML processor software).
2. Insert the resulting string into an attribute as the QCode value.

10.2.1.2.3.3 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 into a Scheme Alias and a Code, identify the first colon, searching from left to right. The string to the left of the colon is the Scheme Alias; the string to the right is the Code. If no colon is found, the QCode is invalid.
4. Check whether the alias is defined in the catalog. If it is not, the QCode is invalid.

10.2.1.2.4 Concept URIs

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

10.2.1.2.4.1 Creating a Concept URI/IRI:

Concatenate the Scheme URI and the Code to obtain the Concept URI.

10.2.1.2.4.2 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 86) **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 by identifiers of a web location from where it can be retrieved, an identifier of the catalog and the version, and an identifier of the authority which manages this catalog. 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. A catalog may be managed by a provider by using a Catalog Item (available only at PCL) - see **Managing Catalogs - catalogItem** (page 59).

It **MUST** have one or more **scheme** (page 288) elements. A scheme element **MUST** have a scheme *alias* attribute and a corresponding scheme *uri* attribute. It **MAY** have a name, a definition and a note element to provide human readable information about the scheme. And the authority governing the scheme **MAY** be indicated by the authority attribute. A **sameAsScheme** (page 286) element **MAY** be used by applying a URI which identifies another scheme with concepts that use the same codes and are semantically equivalent to the concept of this scheme.

Each instance of an 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 270), 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 352) – only support controlled values in the short format of QCodes or full URIs.

Rule for using a QCode (qcode attribute) and a full URI (uri attribute) in property:

- An element **SHOULD NOT** use both a @qcode and a @uri. This rule applies to all properties except conceptId.
- If both attributes, @qcode and @uri, are present the @qcode takes precedence.

A large subset of these properties supports concepts of different types as a value. Therefore typed qualified properties – of datatype **TypedQualPropType** (page 358) – 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 125), **contributor** (page 120) and **infoSource** (page 200) 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 concept identifier (either as QCode or full URI) 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 or uri) and uncontrolled (literal) identifiers - or no identifier at all.

QCodes or URIs on one side (find more about their use in [Qualified and Typed Properties](#) (page 56)) and literals on the other are mutually exclusive for any given property; if one of them exists the other one MUST NOT exist. (The term qcode/uri below indicates that the qcode or the uri or even both attributes can be used to express a controlled value.)

The rules for using the qcode/uri 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/uri 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/uri attribute OR a literal attribute or neither.
- If a literal is used with an assert property then all instances of that literal in that item MUST identify the same concept.
- If a literal is not used with an assert property then it is NOT required that all instances of that literal in that item identify the same concept.

Literals MAY be used in the following cases:

1) As an identifier for linking with an assert element inside a G2 item: 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.

2) 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.

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

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 229](#) on page 307) and Concept Definition Group (see [Table 228](#) on page 307).

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 or 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 323) – 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 81) 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 the Common Power Attributes Group (see **Table 279** on page 361), which includes 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. (Beyond that the group includes more attributes for other purposes.)

11 Managing Catalogs - catalogItem

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

11.1 Description

Catalogs have a quite vital role for all the different NewsML-G2 item types as they provide a key resource for resolving QCodes to URIs identifying concepts: the mapping between scheme URIs and scheme aliases. This is explained in depth in [Processing Catalogs](#) (page 55).

In this context some providers may wish to use the same basic means for managing a Catalog as are available for news content, concepts, editorial planning etc. This purpose is covered by the Catalog Item which has been introduced in NewsML-G2 2.15.

A Catalog Item enables to user to cover this kind of catalog management:

- ◆ A specific set of [Scheme Declaration](#) (page 288) elements is considered to form a catalog.
- ◆ This catalog is made available by a single catalog element which may appear in a stand-alone file as web resource or which may be included into NewsML-G2 items.
- ◆ This catalog element can be made the content of a Catalog Item.
- ◆ The scheme elements of this catalog may be changed (modified or a new one added), but by the general rules of NewsML-G2 this requires to create a new version of the Catalog Item.

11.2 Structure of a Catalog Item

The model of a catalog 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 13) for more information.

11.3 Item Class

The IPTC provides a mandatory standardised scheme applicable to the itemClass property of a catalogItem, identified by the URI <http://cv.iptc.org/newscodes/catinature/>.

11.4 Catalog Related Metadata

The set of metadata related to the catalog-content is listed below. 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 11. Content Metadata Elements

Element Title	Element Name	Card	Described on Page
Date Content Created	contentCreated	(0..1)	129
Date Content Modified	contentModified	(0..1)	130
Creator	creator	(0..unbounded)	125
Contributor	contributor	(0..unbounded)	120
Alternative Identifier	altId	(0..unbounded)	73

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 63) and [Extension Points in XML](#) (page 64).

11.5 Catalog Item Content

A Catalog Item includes a mandatory Catalog Container element which contains a single and mandatory [Catalog](#) (page 86) element as the content of the item.



12 Dealing with Labels and Blocks

12.1 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 76)) for the inclusion of hyperlinks, the [span](#) (page 294) as a generic mechanism for adding information to text, simple [ruby](#) (page 280) markup used in Japanese publications and [inline](#) (page 189) 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 77) property value, plus a basic set of properties defining the concept.

12.2 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 278](#) on page 360) , i.e. an optional language tag (`xml:lang`) and indication of the directionality of textual content (`dir`).

13 Exchanging Items - newsMessage

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

13.1 Description

A **newsMessage** (page 233) 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 207) 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.

13.2 Message Information

All the information about the newsMessage as a wrapper of conveyed G2 items is collected under the **header** (page 220) element which **MUST** be present.

A newsMessage **MUST** have a date of transmission – **sent** (page 142). The date of transmission **MAY** not be updated in case of retransmission of the message.

If 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 and explicitly does **NOT** extend to the children of itemSet.

A newsMessage **MAY** have a **sender** (page 291) 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 300) – 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 258) property to control the overall message transmission process.

It **MAY** indicate the point of **origin** (page 237) of the message, using a provider defined syntax.

It **MAY** have one or more **timestamp** (page 299)(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 151) properties using a provider defined syntax, and the indication of one or more channels – **channel {News Message}** (page 92) – of transmission.

It **MAY** have one or more **signal** (page 292) 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.

13.3 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.





14 Specification Reference

This section provides all specifications for the NewsML-G2 standard (including EventsML-G2), the different specifications tables are cross referenced from other parts of this document.

14.1 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. For NewsML-G2 the names of datatypes end with a "Type" suffix (e.g. QCodeType). Datatypes fall in two groups:
 - Simple data types are primitive data types, as found in software languages or XML schema definitions (e.g.. integer, string). Some restriction may be imposed, such as Int100Type where an integer has been restricted to a value range of 0 to 100.
 - Complex data types are simple data types extended to add further information in order to correctly represent the value. Such ancillary information takes the form of attributes. For example a Label-Type supports mixed content and is extended with language and role attributes.
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.

14.2 General Specifications

14.2.1 XML Namespaces

Table 12. XML Namespace

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



14.2.2 MIME Types

Table 13. IANA Media Types (so called MIME Types)

IANA Media 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.catalogitem+xml	For all kinds of G2 Catalog Items.
application/vnd.iptc.g2.newsmessage+xml	For the G2 News Message

All these Media Types are registered with IANA, see <http://www.iana.org/assignments/media-types/>

14.2.3 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 63).

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 63). 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.

14.2.4 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.

14.3 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 a set of common 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).



14.4 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”.

14.5 Component Structure Format

Table 14 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 14**.

Table 14. 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 <i>nar</i> 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 annotation 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.



Table 14. Component Structure Format

Attribute(s)	One or more XML attributes defined for this component if it is either a complex datatype, an element, or an attribute group.
Child Element(s)	One or more child elements defined for this component if it is either a complex datatype or an element.
XML Schema Note(s)	Any notes regarding the implementation of this component into the XML Schema of this G2-Standard.
Example(s)	One or more XML snippets showing use-cases for this component.



14.6 Element Definitions

14.6.1 Access

Table 15. 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 313)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.2 Access Status

Table 16. 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 350)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.3 Accountable Person

Table 17. 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 334)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.4 Action in Hop History

Table 18. 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 350)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes CV for the target attribute: http://cv.iptc.org/newscodes/hopactiontarget/
Attribute(s)	<ul style="list-style-type: none"> ▪ target (0..1); QCode Type (page 351); targeturi (0..1); IRI Type (page 344); 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)	



14.6.5 Address Line

Table 19. Address Line

(XML) Data Model	Element
Namespace (prefix)	nar
Name	line {address}
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 342)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType; roleuri (0..1); IRIType; Refines the semantics of the property
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.6 Affiliation

Table 20. 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 330)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ timeValidityAttributes (page 361) 	Name	Datatype
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.7 Alternative Identifier

Table 21. 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	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); A qualifier which indicates the context within which the alternative identifier has been allocated. ▪ environment (0..1); QCodeListType (page 349); environmenturi (0..1); IRIType (page 344) A qualifier which indicates the business environment in which the identifier can be used to access the content ▪ idformat (0..1); QCodeType; idformaturi (0..1); IRIType; Indicates the format of the value of the element.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.8 Alternative Locator

Table 22. 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 344)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); A qualifier which indicates the context within which the alternative locator has been allocated. ▪ role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); A refinement of the semantics or business purpose of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.9 Alternative Representation

Table 23. 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 344)																	
Internally Ctrl Values																		
Externally Ctrl Values																		
Attribute(s)	<ul style="list-style-type: none"> ▪ representation (0..1); QCodeType (page 351); representationuri (0..1); IRI Type (page 344); 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; formaturi (0..1); IRIType (page 344); 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. 																	
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
		how (0..1) howuri (0..1)	QCodeType IRIType															
	why (0..1) whyuri (0..1)	QCodeType IRIType																
<ul style="list-style-type: none"> ▪ timeValidityAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>validfrom (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>validto (0..1)</td> <td>DateOptTimeType</td> </tr> </tbody> </table>	Name	Datatype	validfrom (0..1)	DateOptTimeType	validto (0..1)	DateOptTimeType											
	Name	Datatype																
validfrom (0..1)	DateOptTimeType																	
validto (0..1)	DateOptTimeType																	
Child Element(s)																		
XML Schema Note(s)																		
Example(s)																		



14.6.10 Anchor

Table 24. 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 XHTML 1.0 counterpart.		
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ class (0..1); XML Schema String; An equivalent of the HTML class attribute. ▪ href (0..1); IRIType (page 344); 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. 		
	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
	▪ i18nAttributes (page 360)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
	Child Element(s)	<ul style="list-style-type: none"> ▪ span (page 294) (0..unbounded) ▪ ruby (page 280) (0..unbounded) ▪ inline (page 189) (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)			



14.6.11 Assertion

Table 25. 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)	<p>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.</p> <p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 		
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code identifying the concept the assertion is made about. <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI which identifies a concept. <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value identifying the concept the assertion is made about. <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p>		
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
	<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	Name	Datatype
xml:lang (0..1)		XML Schema language	
dir (0..1)		XML Schema string: enumeration <i>ltr, rtl</i> .	



Table 25. Assertion (Continued)

Child Element(s)	<ul style="list-style-type: none"> ▪ 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: <ul style="list-style-type: none"> - 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)	



14.6.12 Assigned To

Table 26. 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 324)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.13 Audience

Table 27. 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 312)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.14 Bag

Table 28. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype	
		id (0..1)	XML Schema ID	
		creator (0..1) creatoruri (0..1)	QCodeType IRIType	
		modified (0..1)	DateOptTimeType	
		custom (0..1)	XML Schema boolean	
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	
		how (0..1) howuri (0..1)	QCodeType IRIType	
		why (0..1) whyuri (0..1)	QCodeType IRIType	
		<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	Name	Datatype
			xml:lang (0..1)	XML Schema language
	dir (0..1)		XML Schema string: enumeration <i>ltr, rtl</i> .	
	Child Element(s)	<ul style="list-style-type: none"> ▪ bit (page 82) (1..unbounded) 		
	XML Schema Note(s)			
Example(s)				



14.6.15 Bag Item

Table 29. 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	QCode Property Type (page 350)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ significance; (0..1); Int100Type (page 340); 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. ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as controlled property value. 		
	▪ quantifyAttributes (page 363)	Name	Datatype
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
		why (0..1) whyuri (0..1)	QCodeType IRIType
derivedFrom (0..1) DEPRECATED	QCodeType		
Child Element(s)			
XML Schema Note(s)			
Note & Example(s)	<p>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.</p> <p>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. <pre><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></pre></p>		



14.6.16 Broader

Table 30. Broader

(XML) Data Model	Element
Namespace (prefix)	nar
Name	broader
Title	Broader
Definition	An identifier of a more generic concept.
User Note(s)	<p><i>rank</i> (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.</p> <p>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.</p> <p>Terms without a rank are treated as if they all have the same rank, which is higher than the rank of all other terms.</p> <p>The same concept may have different ranks in different concept trees. A lower rank results in a placement earlier in a display.</p>
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	RelatedConceptType (page 354)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> 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)	

**14.6.17 By**Table 31. *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 <i>by</i> 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 345)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ rankingAttributes (page 362)	Name rank (0..1)	Datatype XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.18 Capacity

Table 32. 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 345)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.19 Catalog

Table 33. 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)	<ul style="list-style-type: none"> ▪ additionalInfo (0..1); IRIType (page 344); A pointer to some additional information about the Catalog, especially its evolution and latest version. ▪ url (0..1); IRIType (page 344); Defines the location of the catalog as remote resource. (Should be the same as the URL which is used with the href attribute of a catalogRef in an item.) ▪ guid (0..1); XML Schema string; Globally Unique Identifier for this kind of catalog as managed by a provider. A version attribute should be used with it. ▪ version (0..1); XML Schema nonNegativeInteger; Version corresponding to the guid of the catalog. If a version attribute exists a guid attribute must exist too. 		
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
why (0..1) whyuri (0..1)	QCodeType IRIType		
Child Element(s)	<ul style="list-style-type: none"> ▪ title {itemMeta} (page 208) (0..unbounded) ▪ scheme (page 288) (1..unbounded) 		
XML Schema Note(s)			
Example(s)			



14.6.20 Catalog Container

Table 34. Catalog Container

(XML) Data Model	Element
Namespace (prefix)	nar
Name	catalogContainer
Title	Catalog Containeer
Definition	The container of a single catalog.
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)	▪ Catalog (page 86) (1)
XML Schema Note(s)	
Example(s)	



14.6.21 Catalog Item

Table 35. Catalog Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	catalogItem
Title	Catalog Item
Definition	An Item containing a single managed NewsML-G2 catalog
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	AnylItemType (page 310)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	<ul style="list-style-type: none"> ▪ Content Metadata {Catalog Item} (page 106) (0..1) ▪ Catalog Container (page 87) (1)
XML Schema Note(s)	
Example(s)	



14.6.22 Channel of Remote Content

Table 36. Channel of Remote Content

(XML) Data Model	Element
Namespace (prefix)	nar
Name	channel {News Item}
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 36. Channel of Remote Content (Continued)

	<ul style="list-style-type: none"> ▪ chnlid (0..1); XML Schema positiveInteger; A logical identifier of the channel 																																																		
	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The media type of the data conveyed by the channel. 																																																		
	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); The role the data of this channel plays in the scope of the full content. 																																																		
Attribute(s)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>wordcount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>linecount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>pagecount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>width</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>widthunit widthunituri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>height</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>heightunit heightunituri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>orientation</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>layoutorientation layoutorientationuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>colourspace colourspaceuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>colourindicator</td> <td>QCodeType</td> </tr> <tr> <td>colourdepth</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>resolution</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>duration</td> <td>XML Schema string</td> </tr> <tr> <td>durationunit durationunituri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>audiocodec audiocodecuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>audiobitrate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiovbr</td> <td>XML Schema boolean</td> </tr> <tr> <td>audiosamplesize</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiosamplerate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiochannels audiochannelsuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>videocodec videocodecuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>videoavgbitrate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>videovbr</td> <td>XML Schema boolean</td> </tr> </tbody> </table>	Name	Datatype	wordcount	XML Schema nonNegativeInteger	linecount	XML Schema nonNegativeInteger	pagecount	XML Schema nonNegativeInteger	width	XML Schema nonNegativeInteger	widthunit widthunituri	QCodeType IRIType	height	XML Schema nonNegativeInteger	heightunit heightunituri	QCodeType IRIType	orientation	XML Schema nonNegativeInteger	layoutorientation layoutorientationuri	QCodeType IRIType	colourspace colourspaceuri	QCodeType IRIType	colourindicator	QCodeType	colourdepth	XML Schema nonNegativeInteger	resolution	XML Schema positiveInteger	duration	XML Schema string	durationunit durationunituri	QCodeType IRIType	audiocodec audiocodecuri	QCodeType IRIType	audiobitrate	XML Schema positiveInteger	audiovbr	XML Schema boolean	audiosamplesize	XML Schema positiveInteger	audiosamplerate	XML Schema positiveInteger	audiochannels audiochannelsuri	QCodeType IRIType	videocodec videocodecuri	QCodeType IRIType	videoavgbitrate	XML Schema positiveInteger	videovbr	XML Schema boolean
	Name	Datatype																																																	
	wordcount	XML Schema nonNegativeInteger																																																	
	linecount	XML Schema nonNegativeInteger																																																	
	pagecount	XML Schema nonNegativeInteger																																																	
	width	XML Schema nonNegativeInteger																																																	
	widthunit widthunituri	QCodeType IRIType																																																	
	height	XML Schema nonNegativeInteger																																																	
	heightunit heightunituri	QCodeType IRIType																																																	
	orientation	XML Schema nonNegativeInteger																																																	
	layoutorientation layoutorientationuri	QCodeType IRIType																																																	
	colourspace colourspaceuri	QCodeType IRIType																																																	
	colourindicator	QCodeType																																																	
	colourdepth	XML Schema nonNegativeInteger																																																	
	resolution	XML Schema positiveInteger																																																	
	duration	XML Schema string																																																	
	durationunit durationunituri	QCodeType IRIType																																																	
	audiocodec audiocodecuri	QCodeType IRIType																																																	
	audiobitrate	XML Schema positiveInteger																																																	
	audiovbr	XML Schema boolean																																																	
	audiosamplesize	XML Schema positiveInteger																																																	
	audiosamplerate	XML Schema positiveInteger																																																	
	audiochannels audiochannelsuri	QCodeType IRIType																																																	
videocodec videocodecuri	QCodeType IRIType																																																		
videoavgbitrate	XML Schema positiveInteger																																																		
videovbr	XML Schema boolean																																																		
	<ul style="list-style-type: none"> ▪ newsContentCharacteristics (page 365) 																																																		



Table 36. Channel of Remote Content (Continued)

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



14.6.23 Channel for News Message

Table 37. Channel for News Message

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	channel {News Message}		
Title	Channel for News Message		
Definition	A transmission channel used by the message.		
User Note(s)	<p>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.</p> <p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 		
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 363)	Name	Datatype
		qcode (0..1)	QCodeType
		uri (0..1)	XML Schema anyURI
		literal (0..1)	NormalizedStringType (of G2)
		type typeuri	QCodeType IRIType
role roleuri	QCodeType IRIType		
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.24 Circle

Table 38. Circle

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	circle		
Title	Circle as geoArea		
Definition	Definition of a circular geometry as a geographic area.		
User Note(s)	The position element defines the centre of the circle		
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
		<ul style="list-style-type: none"> ▪ radius (1); XML Schema double; The radius of the circle ▪ radunit (0..1); QCodeType; radunituri (0..1); IRIType; The dimension unit of the radius. The use of either the radunit or the radunituri attribute is mandatory (from NewsML-G2 v 2.19 onwards) 	
Child Element(s)	<ul style="list-style-type: none"> ▪ position (page 172) (1) 		
XML Schema Note(s)			
Example(s)	<pre><geoAreaDetails> <circle radius="1.335" radunit="dimensionunit:km"> <position ...> </circle> </geoAreaDetail></pre>		



14.6.25 Concept

Table 39. 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)	<ul style="list-style-type: none"> id (0..1); XML Schema ID; The local identifier of the property 		
	<ul style="list-style-type: none"> i18nAttributes (page 360) 	Name	Datatype
		xml:lang (0..1)	XML Schema language
	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl.</i>	



Table 39. Concept (Continued)

Child Element(s)	▪ conceptId (page 98) (1)		
	▪ type (page 301) (0..1)		
	▪ name (page 100) (1..unbounded)		
	▪ definition (page 96) (0..unbounded)		
	▪ note (page 101) (0..unbounded)		
	▪ facet (page 165) (0..unbounded)		
	▪ remoteInfo (page 275) (0..unbounded)		
	▪ hierarchyInfo (page 183) (0..unbounded)		
	▪ Concept Relationships Group (page 307) (0..1)	Element Name	Page
		broader (0..unbounded)	83
		narrower (0..unbounded)	222
		related (0..unbounded)	267
	▪ Entity Details Group (page 307) (0..1)	Element Name	Page
		geoAreaDetails (1)	173
		organisationDetails (1)	238
personDetails (1)		248	
POIDetails (1)		253	
objectDetails (1)		234	
▪ eventDetails (1)		158	
▪ conceptExtProperty (page 97) (0..unbounded)			
▪ eventDetails (page 158) (0..1)			
▪ Extension Point (0..1). Any set of provider-defined properties.			
XML Schema Note(s)			
Example(s)			



14.6.26 Concept Definition

Table 40. 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 313)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> timeValidityAttributes (page 361) 	Name	Datatype
		validfrom (0..1)	DateOptTimeType
	<ul style="list-style-type: none"> roleuri 0..1 ; IRIType (page 344); A URI identifying the refined semantics of the name. 	validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.27 Concept Extension Property

Table 41. Concept Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	conceptExtProperty
Title	Concept Extension Property
Definition	A generic extension property for a concept; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.28 Concept Identifier

Table 42. 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 315)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.29 Concept Item

Table 43. 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	AnylItemType (page 310)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ contentMeta {Concept} (page 107) (0..1)
	▪ assert (page 77) (0..unbounded)
	▪ inlineRef (page 194) (0..unbounded)
	▪ derivedFrom (page 150) (0..unbounded)
	▪ concept (page 94) (0..1)
XML Schema Note(s)	
Example(s)	



14.6.30 Concept Name

Table 44. 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 342).							
Internally Ctrl Values								
Externally Ctrl Values	Recommended IPTC NewsCodes CV for the <i>part</i> attribute: http://cv.iptc.org/newscodes/namepart/							
Attribute(s)	<ul style="list-style-type: none"> role (0..1); QCodeListType (page 349); roleuri (0..1); IRI List Type (page 343); A refinement of the semantics of the name. 							
	<ul style="list-style-type: none"> roleuri 0..1; IRIType (page 344); A URI identifying the refined semantics of the name. 							
	<ul style="list-style-type: none"> part (0..1); QCodeType (page 351); parturi (0..1); IRI Type (page 344); Specifies which part of a full name this property provides. 							
		timeValidityAttributes (page 361)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>validfrom (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>validto (0..1)</td> <td>DateOptTimeType</td> </tr> </tbody> </table>	Name	Datatype	validfrom (0..1)	DateOptTimeType	validto (0..1)
Name	Datatype							
validfrom (0..1)	DateOptTimeType							
validto (0..1)	DateOptTimeType							
Child Element(s)								
XML Schema Note(s)								
Example(s)								



14.6.31 Concept Note

Table 45. Note

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	note		
Title	Concept Note		
Definition	Additional natural-language information about the concept.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	BlockType (page 313)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ timeValidityAttributes (page 361)	Name	Datatype
		validfrom (0..1)	DateOptTimeType
	validto (0..1)	DateOptTimeType	
	▪ roleuri 0..1; IRIType (page 344); A URI identifying the refined semantics of the note.		
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.32 Concept Reference

Table 46. 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 323)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.33 Concept Set

Table 47. 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 94) (0..unbounded)
XML Schema Note(s)	
Example(s)	



14.6.34 Confirmation

Table 48. 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 350)
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)	



14.6.35 Contact Information

Table 49. 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 <i>role</i> 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/eventcontactinfole/
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); ▪ roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the set of contact information.
Child Element(s)	<ul style="list-style-type: none"> ▪ email (page 155) (0..unbounded) ▪ im (page 199) (0..unbounded) ▪ phone (page 249) (0..unbounded) ▪ fax (page 166) (0..unbounded) ▪ web (page 306) (0..unbounded) ▪ address (page 255) (0..unbounded) ▪ note (page 101) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	



14.6.36 Content Metadata {Catalog Item}

Table 50. Content Metadata for a Catalog Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta {Catalog}		
Title	Content Metadata for a Catalog Item		
Definition	A set of metadata properties about the content of a Catalog Item.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
Child Element(s)	<ul style="list-style-type: none"> ▪ Administrative Metadata Group (page 308) (0..1) 	Element Name	Page
		contributor (0..unbounded)	120
		creator (0..unbounded)	125
		contentCreated (0..1)	129
		contentModified (0..1)	130
		altId (0..unbounded)	73
	<ul style="list-style-type: none"> ▪ contentMetaExtProperty (page 117) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 		
XML Schema Note(s)			
Example(s)			



14.6.37 Content Metadata {Concept Item}

Table 51. 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 360)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .



Table 51. Content Metadata for a Concept Item

	<ul style="list-style-type: none"> ▪ icon (page 186) (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. 																											
Child Element(s)	<ul style="list-style-type: none"> ▪ Administrative Metadata Group (page 308) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>audience (0..unbounded)</td> <td>80</td> </tr> <tr> <td>contributor (0..unbounded)</td> <td>120</td> </tr> <tr> <td>creator (0..unbounded)</td> <td>125</td> </tr> <tr> <td>contentCreated (0..1)</td> <td>129</td> </tr> <tr> <td>contentModified (0..1)</td> <td>130</td> </tr> <tr> <td>located (0..unbounded)</td> <td>216</td> </tr> <tr> <td>infoSource (0..unbounded)</td> <td>200</td> </tr> <tr> <td>urgency (0..1)</td> <td>303</td> </tr> <tr> <td>exclAudience (0..unbounded)</td> <td>161</td> </tr> <tr> <td>altId (0..unbounded)</td> <td>73</td> </tr> <tr> <td>rating (page 262) (0..unbounded)</td> <td></td> </tr> <tr> <td>userInteraction (page 302) (0..unbounded)</td> <td></td> </tr> </tbody> </table>	Element Name	Page	audience (0..unbounded)	80	contributor (0..unbounded)	120	creator (0..unbounded)	125	contentCreated (0..1)	129	contentModified (0..1)	130	located (0..unbounded)	216	infoSource (0..unbounded)	200	urgency (0..1)	303	exclAudience (0..unbounded)	161	altId (0..unbounded)	73	rating (page 262) (0..unbounded)		userInteraction (page 302) (0..unbounded)	
		Element Name	Page																									
		audience (0..unbounded)	80																									
		contributor (0..unbounded)	120																									
		creator (0..unbounded)	125																									
		contentCreated (0..1)	129																									
		contentModified (0..1)	130																									
		located (0..unbounded)	216																									
		infoSource (0..unbounded)	200																									
		urgency (0..1)	303																									
		exclAudience (0..unbounded)	161																									
		altId (0..unbounded)	73																									
		rating (page 262) (0..unbounded)																										
		userInteraction (page 302) (0..unbounded)																										
<ul style="list-style-type: none"> ▪ Descriptive Metadata Core Group (page 308) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>description (0..unbounded)</td> <td>146</td> </tr> <tr> <td>headline (0..unbounded)</td> <td>182</td> </tr> <tr> <td>keyword (0..unbounded)</td> <td>209</td> </tr> <tr> <td>language (0..unbounded)</td> <td>211</td> </tr> <tr> <td>slugline (0..unbounded)</td> <td>293</td> </tr> <tr> <td>subject (0..unbounded)</td> <td>296</td> </tr> </tbody> </table>	Element Name	Page	description (0..unbounded)	146	headline (0..unbounded)	182	keyword (0..unbounded)	209	language (0..unbounded)	211	slugline (0..unbounded)	293	subject (0..unbounded)	296													
	Element Name	Page																										
	description (0..unbounded)	146																										
	headline (0..unbounded)	182																										
	keyword (0..unbounded)	209																										
	language (0..unbounded)	211																										
slugline (0..unbounded)	293																											
subject (0..unbounded)	296																											
<ul style="list-style-type: none"> ▪ contentMetaExtProperty (page 117) (0..unbounded) 																												
<ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 																												
XML Schema Note(s)																												
Example(s)																												



14.6.38 Content Metadata {Knowledge Item}

Table 52. 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 360)	Name	Datatype	
		xml:lang (0..1)	XML Schema language	
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .	
Child Element(s)	▪ icon (page 186) (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.			
			Element Name	Page
			audience (0..1)	80
			contributor (0..unbounded)	120
			creator (0..unbounded)	125
			contentCreated (0..1)	129
			contentModified (0..1)	130
			located (0..unbounded)	216
			infoSource (0..unbounded)	200
			urgency (0..1)	303
			exclAudience (0..unbounded)	161
			altId (0..unbounded)	73
			rating (page 262) (0..unbounded)	
			userInteraction (page 302) (0..unbounded)	
				Element Name
		description (0..unbounded)	146	
		headline (0..unbounded)	182	
		keyword (0..unbounded)	209	
		language (0..unbounded)	211	
		slugline (0..unbounded)	293	
		subject (0..unbounded)	296	
	▪ contentMetaExtProperty (page 117) (0..unbounded)			
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.			



Table 52. Content Metadata for a Knowledge Item (Continued)

XML Schema Note(s)	
Example(s)	



14.6.39 Content Metadata {News Item}

Table 53. 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)	<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .



Table 53. Content Metadata for a News Item (Continued)

	<ul style="list-style-type: none"> ▪ icon (page 186) (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. 																							
Child Element(s)	<ul style="list-style-type: none"> ▪ Administrative Metadata Group (page 308) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>audience (0..unbounded)</td> <td>80</td> </tr> <tr> <td>contributor (0..unbounded)</td> <td>120</td> </tr> <tr> <td>creator (0..unbounded)</td> <td>125</td> </tr> <tr> <td>contentCreated (0..1)</td> <td>129</td> </tr> <tr> <td>contentModified (0..1)</td> <td>130</td> </tr> <tr> <td>located (0..unbounded)</td> <td>216</td> </tr> <tr> <td>infoSource (0..unbounded)</td> <td>200</td> </tr> <tr> <td>urgency (0..1)</td> <td>303</td> </tr> <tr> <td>exclAudience (0..unbounded)</td> <td>161</td> </tr> <tr> <td>altId (0..unbounded)</td> <td>73</td> </tr> </tbody> </table>	Element Name	Page	audience (0..unbounded)	80	contributor (0..unbounded)	120	creator (0..unbounded)	125	contentCreated (0..1)	129	contentModified (0..1)	130	located (0..unbounded)	216	infoSource (0..unbounded)	200	urgency (0..1)	303	exclAudience (0..unbounded)	161	altId (0..unbounded)	73
		Element Name	Page																					
		audience (0..unbounded)	80																					
		contributor (0..unbounded)	120																					
		creator (0..unbounded)	125																					
		contentCreated (0..1)	129																					
		contentModified (0..1)	130																					
		located (0..unbounded)	216																					
		infoSource (0..unbounded)	200																					
		urgency (0..1)	303																					
		exclAudience (0..unbounded)	161																					
		altId (0..unbounded)	73																					
			rating (page 262) (0..unbounded)																					
		userInteraction (page 302) (0..unbounded)																						
	<ul style="list-style-type: none"> ▪ Descriptive Metadata Group (page 309) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>by (0..unbounded)</td> <td>84</td> </tr> <tr> <td>creditline (0..unbounded)</td> <td>126</td> </tr> <tr> <td>dateline (0..unbounded)</td> <td>143</td> </tr> <tr> <td>description (0..unbounded)</td> <td>146</td> </tr> <tr> <td>genre (0..unbounded)</td> <td>171</td> </tr> <tr> <td>headline (0..unbounded)</td> <td>182</td> </tr> <tr> <td>keyword (0..unbounded)</td> <td>209</td> </tr> <tr> <td>language (0..unbounded)</td> <td>211</td> </tr> <tr> <td>slugline (0..unbounded)</td> <td>293</td> </tr> <tr> <td>subject (0..unbounded)</td> <td>296</td> </tr> </tbody> </table>	Element Name	Page	by (0..unbounded)	84	creditline (0..unbounded)	126	dateline (0..unbounded)	143	description (0..unbounded)	146	genre (0..unbounded)	171	headline (0..unbounded)	182	keyword (0..unbounded)	209	language (0..unbounded)	211	slugline (0..unbounded)	293	subject (0..unbounded)	296
		Element Name	Page																					
		by (0..unbounded)	84																					
		creditline (0..unbounded)	126																					
		dateline (0..unbounded)	143																					
		description (0..unbounded)	146																					
genre (0..unbounded)		171																						
headline (0..unbounded)		182																						
keyword (0..unbounded)		209																						
language (0..unbounded)		211																						
slugline (0..unbounded)	293																							
subject (0..unbounded)	296																							
	▪ contentMetaExtProperty (page 117) (0..unbounded)																							
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.																							
XML Schema Note(s)																								
Example(s)																								



14.6.40 Content Metadata {Package Item}

Table 54. 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)	<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .



Table 54. Content Metadata for a Package Item (Continued)

	<ul style="list-style-type: none"> ▪ icon (page 186) (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. 																							
Child Element(s)	<ul style="list-style-type: none"> ▪ Administrative Metadata Group (page 308) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>audience (0..unbounded)</td> <td>80</td> </tr> <tr> <td>contributor (0..unbounded)</td> <td>120</td> </tr> <tr> <td>creator (0..unbounded)</td> <td>125</td> </tr> <tr> <td>contentCreated (0..1)</td> <td>129</td> </tr> <tr> <td>contentModified (0..1)</td> <td>130</td> </tr> <tr> <td>located (0..unbounded)</td> <td>216</td> </tr> <tr> <td>infoSource (0..unbounded)</td> <td>200</td> </tr> <tr> <td>urgency (0..1)</td> <td>303</td> </tr> <tr> <td>exclAudience (0..unbounded)</td> <td>161</td> </tr> <tr> <td>altId (0..unbounded)</td> <td>73</td> </tr> </tbody> </table>	Element Name	Page	audience (0..unbounded)	80	contributor (0..unbounded)	120	creator (0..unbounded)	125	contentCreated (0..1)	129	contentModified (0..1)	130	located (0..unbounded)	216	infoSource (0..unbounded)	200	urgency (0..1)	303	exclAudience (0..unbounded)	161	altId (0..unbounded)	73
		Element Name	Page																					
		audience (0..unbounded)	80																					
		contributor (0..unbounded)	120																					
		creator (0..unbounded)	125																					
		contentCreated (0..1)	129																					
		contentModified (0..1)	130																					
		located (0..unbounded)	216																					
		infoSource (0..unbounded)	200																					
		urgency (0..1)	303																					
		exclAudience (0..unbounded)	161																					
		altId (0..unbounded)	73																					
		<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>rating (page 262) (0..unbounded)</td> <td></td> </tr> <tr> <td>userInteraction (page 302) (0..unbounded)</td> <td></td> </tr> </tbody> </table>	Element Name	Page	rating (page 262) (0..unbounded)		userInteraction (page 302) (0..unbounded)																	
	Element Name	Page																						
	rating (page 262) (0..unbounded)																							
	userInteraction (page 302) (0..unbounded)																							
	<ul style="list-style-type: none"> ▪ Descriptive Metadata Group (page 309) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>by (0..unbounded)</td> <td>84</td> </tr> <tr> <td>creditline (0..unbounded)</td> <td>126</td> </tr> <tr> <td>dateline (0..unbounded)</td> <td>143</td> </tr> <tr> <td>description (0..unbounded)</td> <td>146</td> </tr> <tr> <td>genre (0..unbounded)</td> <td>171</td> </tr> <tr> <td>headline (0..unbounded)</td> <td>182</td> </tr> <tr> <td>keyword (0..unbounded)</td> <td>209</td> </tr> <tr> <td>language (0..unbounded)</td> <td>211</td> </tr> <tr> <td>slugline (0..unbounded)</td> <td>293</td> </tr> <tr> <td>subject (0..unbounded)</td> <td>296</td> </tr> </tbody> </table>	Element Name	Page	by (0..unbounded)	84	creditline (0..unbounded)	126	dateline (0..unbounded)	143	description (0..unbounded)	146	genre (0..unbounded)	171	headline (0..unbounded)	182	keyword (0..unbounded)	209	language (0..unbounded)	211	slugline (0..unbounded)	293	subject (0..unbounded)	296
		Element Name	Page																					
		by (0..unbounded)	84																					
		creditline (0..unbounded)	126																					
dateline (0..unbounded)		143																						
description (0..unbounded)		146																						
genre (0..unbounded)		171																						
headline (0..unbounded)		182																						
keyword (0..unbounded)		209																						
language (0..unbounded)		211																						
slugline (0..unbounded)	293																							
subject (0..unbounded)	296																							
	<ul style="list-style-type: none"> ▪ contentMetaExtProperty (page 117) (0..unbounded) 																							
	<ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 																							
XML Schema Note(s)																								
Example(s)																								



14.6.41 Content Metadata {Planning Item}

Table 55. 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 360)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .



Table 55. Content Metadata for a Planning Item (Continued)

	<ul style="list-style-type: none"> ▪ icon (page 186) (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. 																											
Child Element(s)	<ul style="list-style-type: none"> ▪ Administrative Metadata Group (page 308) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>audience (0..unbounded)</td> <td>80</td> </tr> <tr> <td>contributor (0..unbounded)</td> <td>120</td> </tr> <tr> <td>creator (0..unbounded)</td> <td>125</td> </tr> <tr> <td>contentCreated (0..1)</td> <td>129</td> </tr> <tr> <td>contentModified (0..1)</td> <td>130</td> </tr> <tr> <td>located (0..unbounded)</td> <td>216</td> </tr> <tr> <td>infoSource (0..unbounded)</td> <td>200</td> </tr> <tr> <td>urgency (0..1)</td> <td>303</td> </tr> <tr> <td>exclAudience (0..unbounded)</td> <td>161</td> </tr> <tr> <td>altId (0..unbounded)</td> <td>73</td> </tr> <tr> <td>rating (page 262) (0..unbounded)</td> <td></td> </tr> <tr> <td>userInteraction (page 302) (0..unbounded)</td> <td></td> </tr> </tbody> </table>	Element Name	Page	audience (0..unbounded)	80	contributor (0..unbounded)	120	creator (0..unbounded)	125	contentCreated (0..1)	129	contentModified (0..1)	130	located (0..unbounded)	216	infoSource (0..unbounded)	200	urgency (0..1)	303	exclAudience (0..unbounded)	161	altId (0..unbounded)	73	rating (page 262) (0..unbounded)		userInteraction (page 302) (0..unbounded)	
		Element Name	Page																									
		audience (0..unbounded)	80																									
		contributor (0..unbounded)	120																									
		creator (0..unbounded)	125																									
		contentCreated (0..1)	129																									
		contentModified (0..1)	130																									
		located (0..unbounded)	216																									
		infoSource (0..unbounded)	200																									
		urgency (0..1)	303																									
		exclAudience (0..unbounded)	161																									
		altId (0..unbounded)	73																									
		rating (page 262) (0..unbounded)																										
		userInteraction (page 302) (0..unbounded)																										
<ul style="list-style-type: none"> ▪ Descriptive Metadata Core Group (page 308) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>description (0..unbounded)</td> <td>146</td> </tr> <tr> <td>headline (0..unbounded)</td> <td>182</td> </tr> <tr> <td>keyword (0..unbounded)</td> <td>209</td> </tr> <tr> <td>language (0..unbounded)</td> <td>211</td> </tr> <tr> <td>slugline (0..unbounded)</td> <td>293</td> </tr> <tr> <td>subject (0..unbounded)</td> <td>296</td> </tr> </tbody> </table>	Element Name	Page	description (0..unbounded)	146	headline (0..unbounded)	182	keyword (0..unbounded)	209	language (0..unbounded)	211	slugline (0..unbounded)	293	subject (0..unbounded)	296													
	Element Name	Page																										
	description (0..unbounded)	146																										
	headline (0..unbounded)	182																										
	keyword (0..unbounded)	209																										
	language (0..unbounded)	211																										
slugline (0..unbounded)	293																											
subject (0..unbounded)	296																											
<ul style="list-style-type: none"> ▪ contentMetaExtProperty (page 117) (0..unbounded) 																												
<ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 																												
XML Schema Note(s)																												
Example(s)																												



14.6.42 Content Metadata Extension Property

Table 56. Content Metadata Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	contentMetaExtProperty
Title	Content Metadata Extension Property
Definition	A generic extension property for content metadata; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.43 Content Provider

Table 57. 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 332)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.44 Content Set

Table 58. 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)	<ul style="list-style-type: none"> ▪ 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)	<ul style="list-style-type: none"> ▪ inlineXML (page 195) (0..unbounded) Or ▪ inlineData (page 191) (0..unbounded) Or ▪ remoteContent (page 271) (0..unbounded)
XML Schema Note(s)	
Example(s)	



14.6.45 Contributor

Table 59. 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 332)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the property. ▪ jobtitle (0..1); QCodeType (page 351); jobtitleuri (0..1); IRI Type (page 344) The job title of the creator in the news provider organisation.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.46 Copyright Holder

Table 60. 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 332)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.47 Copyright Notice

Table 61. 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 355)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.48 Country

Table 62. 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 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.49 Country Area

Table 63. 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 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCode List Type (page 349); roleuri (0..1); IRIListType (page 343); Refines the semantics of the property
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.50 Creator

Table 64. 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 332)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the property. ▪ jobtitle (0..1); QCodeType (page 351); jobtitleuri (0..1); IRI Type (page 344); The job title of the creator in the news provider organisation.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.51 Credit Line

Table 65. 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 362)	Name	Datatype
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.52 Date an Object Ceased to Exist

Table 66. *Date an Object ceased to Exist*

(XML) Data Model	Element
Namespace (prefix)	nar
Name	ceasedToExist {Object}
Title	<i>Date an Object ceased to Exist</i>
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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.53 Date a Point Of Interest Ceased to Exist

Table 67. Date a Point Of Interest ceased to Exist

(XML) Data Model	Element
Namespace (prefix)	nar
Name	ceasedToExist {POI}
Title	<i>Date a Point of Interest ceased to Exist</i>
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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.54 Date Content Created

Table 68. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.55 Date Content Modified

Table 69. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.56 Date Item Embargo Ends

Table 70. Date Item Embargo Ends

(XML) Data Model	Element
Namespace (prefix)	nar
Name	embargoed
Title	Date Item Embargo Ends
Definition	<p>The date and time (with the time zone) before which all versions of the Item are embargoed.</p> <p>If the element is absent, the Item is not embargoed.</p> <p>If the element exists but is empty the end of the embargo is defined by the language in an edNote (page 153) element.</p>
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	DateTimeOrNullPropType (page 319)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.57 Date Item First Created

Table 71. 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 320)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.58 Date Item Version Created

Table 72. 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 320)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.59 Date of Birth of Person

Table 73. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.60 Date of Creation of Object

Table 74. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.61 Date of Creation of Point Of Interest

Table 75. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.62 Date of Death of Person

Table 76. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.63 Date of Dissolution of Geopolitical Area

Table 77. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.64 Date of Dissolution of Organisation

Table 78. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.65 Date of Foundation of Geopolitical Area

Table 79. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.66 Date of Foundation of Organisation

Table 80. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.67 Date of Transmission

Table 81. 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 320)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.68 Dateline

Table 82. 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)	<p>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.</p> <p>Traditionally a dateline indicates when and where news content is created, not necessarily the time and place relative to the news event.</p> <p>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).</p>						
Implementation Note(s)							
XML Schema Spec	At: PCL						
Datatype	Label1Type (page 345)						
Internally Ctrl Values							
Externally Ctrl Values							
Attribute(s)	<ul style="list-style-type: none"> ▪ rankingAttributes (page 362) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>rank (0..1)</td> <td>XML Schema nonNegativeInteger</td> </tr> </tbody> </table>	Name	Datatype	rank (0..1)	XML Schema nonNegativeInteger	
Name	Datatype						
rank (0..1)	XML Schema nonNegativeInteger						
Child Element(s)							
XML Schema Note(s)							
Example(s)							



14.6.69 Dates

Table 83. 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 295) (1)
	▪ end (page 156) (0..1) Or
	▪ duration (page 152) (0..1)
	▪ confirmation (page 104) (0..1)
	▪ Recurrence Group (see Table 84) (0..1)
XML Schema Note(s)	
Example(s)	

14.6.70 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 84. Recurrence Group Elements

Element Title	Element Name	Card	Described on Page
Recurrence Date	rDate	(0..unbounded)	264
Recurrence Rule	rRule	(0..unbounded)	265
Exclusion Date	exDate	(0..unbounded)	162
Exclusion Rule	exRule	(0..unbounded)	163



14.6.71 Date Resource Created

Table 85. 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 356)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.72 Description

Table 86. 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 313)		
Internally Ctrl Values			
Externally Ctrl Values	Recommended IPTC NewsCodes for the <i>role</i> attribute: http://cv.iptc.org/newscodes/descriptionrole/		
Attribute(s)	▪ rankingAttributes (page 362)	Name rank (0..1)	Datatype XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.73 Deliverable Of

Table 87. 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 346)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.74 Delivered Item Reference

Table 88. 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 346)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.75 Delivery Information

Table 89. 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: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	<ul style="list-style-type: none"> ▪ deliveredItemRef (page 148) (1..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	



14.6.76 Derived From

Table 90. Derived From

(XML) Data Model	Element
Namespace (prefix)	nar
Name	derivedFrom
Title	Derived From
Definition	Represents a concept which was used for deriving the value of a property in this G2 item.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flex1PropType (page 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ idrefs (1); XML Schema IDREFS; Refers to the ids of elements which values have been derived from the concept represented by this property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.77 Destination

Table 91. Destination

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	destination		
Title	Destination		
Definition	The point(s) of destination of the message.		
User Note(s)	<p>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”.</p> <p>The structure of this string is not specified by the IPTC.</p> <p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 		
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 363)	Name	Datatype
		qcode (0..1)	QCodeType
		uri (0..1)	XML Schema anyURI
		literal (0..1)	NormalizedStringType (of G2)
		type typeuri	QCodeType IRIType
		role roleuri	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.78 Duration

Table 92. 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 295) 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)	



14.6.79 Editorial Note

Table 93. 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 313)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.80 Editorial Service

Table 94. 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 352)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.81 Email Address

Table 95. 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 321)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.82 End Date/Time

Table 96. 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 311)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.83 Event

Table 97. 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)	<ul style="list-style-type: none"> ▪ eventDetails (page 158) (1) ▪ name (page 100) (1..unbounded) ▪ definition (page 96) (0..unbounded) ▪ facet (page 165) (0..unbounded) ▪ note (page 101) (0..unbounded) 		
	<ul style="list-style-type: none"> ▪ Concept Relationships Group (page 307) (1) 	Element Name	Page
		broader (0..unbounded)	83
		narrower (0..unbounded)	222
		related (0..unbounded)	267
		sameAs {Relationship} (0..unbounded)	284
XML Schema Note(s)			
Example(s)			



14.6.84 Event Details

Table 98. 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)	<ul style="list-style-type: none"> ▪ dates (page 144) (1) ▪ occurStatus (page 235) (0..1) ▪ newsCoverageStatus (page 231) (0..1) ▪ registration (page 266) (1..unbounded) ▪ accessStatus (page 68) (0..unbounded) ▪ subject (page 296) (0..unbounded); ▪ location (page 159) (0..unbounded) ▪ participant (page 242) (0..unbounded) ▪ participationRequirement (page 243) (0..unbounded) ▪ organiser (page 239) (0..unbounded) ▪ contactInfo (page 105) (0..unbounded) ▪ language (page 211) (0..unbounded) ▪ newsCoverage {Concept} (page 226) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	



14.6.85 Event Location

Table 99. 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 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); A refinement on the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.86 Events Wrapper

Table 100. 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 157) (1..unbounded)
XML Schema Note(s)	
Example(s)	



14.6.87 Excluded Audience

Table 101. 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 312)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.88 Exclusion Date

Table 102. 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 317)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.89 Exclusion Rule

Table 103. 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 353)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.90 Expiry Date

Table 104. Expiry Date

(XML) Data Model	Element
Namespace (prefix)	nar
Name	expires
Title	Expiry Date
Definition	The date and time after which the Newsletter is no longer considered valid by its publisher.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	Date and Optional Time Property Type (page 317)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.91 Facet (DEPRECATED)

Table 105. 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 336)		
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)	<ul style="list-style-type: none"> ▪ rel (0..1); QCodeType (page 351); reluri (0..1); IRI Type (page 344); The identifier of the relationship between the current concept (containing the facet) and the concept identified by the facet value. The default value for <i>rel</i> is the "IsA" relationship, this applies also if the <i>rel</i> attribute is omitted.		
	▪ timeValidityAttributes (page 361)	Name	Datatype
		validfrom (0..1)	DateOptTimeType
	validto (0..1)	DateOptTimeType	
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.92 Fax Number

Table 106. 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 321)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.93 File Name

Table 107. 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	Normalized String Type (page 349)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.94 G2 Content Type

Table 108. 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 64) types like application/vnd.iptc.g2.*item+xml. See: http://www.iana.org/assignments/media-types/application/		
Externally Ctrl Values			
Attribute(s)	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.95 G2 Item Class

Table 109. 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 352)
Internally Ctrl Values	Any of the G2-Standards specific IANA MIME (see MIME Types on page 64) 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)	



14.6.96 Generator Tool

Table 110. 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 359)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); Identifies the stage at which this generator was used.
Child Element(s)	
XML Schema Note(s)	
Example(s)	<pre> <!-- 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> </pre>



14.6.97 Genre

Table 111. 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 323)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ rankingAttributes (page 362)	Name	Datatype
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.98 Geographic Position

Table 112. Geographic Position

(XML) Data Model	Element
Namespace (prefix)	nar
Name	position
Title	Geographic Position
Definition	The geographic coordinates of the location.
User Note(s)	<p>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 <i>gpsdatum</i> attribute, WGS84 is the default system.</p>
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ 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)	<ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

14.6.99 Geopolitical Area Details

Table 113. 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)	<ul style="list-style-type: none"> ▪ position (page 172) (0..1) ▪ line {geoArea} (page 212) (0..unbounded) ▪ circle (page 93) (0..unbounded) ▪ polygon (page 254) (0..unbounded) ▪ founded {geoArea} (page 140) (0..1) ▪ dissolved {geoArea} (page 138) (0..1) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	<pre> <geoAreaDetails> <line> <position ... > <position ...> </line> <circle radius="1.335" radunit="dimensionunit:km"> <position ...> </circle> </geoAreaDetail> </pre>



14.6.100 Group

Table 114. Group

(XML) Data Model	Element
Namespace (prefix)	nar
Name	group
Title	Group
Definition	A mixed set of group references and links.
User Note(s)	<ul style="list-style-type: none"> ▪ Group Mode: By default the group is “complementary and unordered”. <ul style="list-style-type: none"> ▪ 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	



Table 114. Group (Continued)

	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); <p>The part this group plays within its container. The use of either the role or the roleuri attribute is mandatory (from NewsML-G2 v 2.19 onwards)</p>																									
	<ul style="list-style-type: none"> ▪ mode (0..unbounded); QCodeType (page 351); modeuri (0..unbounded); IRI Type (page 344); <p>An indication whether the elements in the group are complementary and unordered, complementary and ordered or a set of alternative elements.</p>																									
Attribute(s)	<table border="1"> <thead> <tr> <th data-bbox="432 562 842 600"></th> <th data-bbox="842 562 1106 600">Name</th> <th data-bbox="1106 562 1444 600">Datatype</th> </tr> </thead> <tbody> <tr> <td data-bbox="432 600 842 645" rowspan="10">▪ commonPowerAttributes (page 361)</td> <td data-bbox="842 600 1106 645">id (0..1)</td> <td data-bbox="1106 600 1444 645">XML Schema ID</td> </tr> <tr> <td data-bbox="842 645 1106 719">creator (0..1) creatoruri (0..1)</td> <td data-bbox="1106 645 1444 719">QCodeType IRIType</td> </tr> <tr> <td data-bbox="842 719 1106 763">modified (0..1)</td> <td data-bbox="1106 719 1444 763">DateOptTimeType</td> </tr> <tr> <td data-bbox="842 763 1106 808">custom (0..1)</td> <td data-bbox="1106 763 1444 808">XML Schema boolean</td> </tr> <tr> <td data-bbox="842 808 1106 913">pubconstraint (0..1) pubconstrainturi (0..1)</td> <td data-bbox="1106 808 1444 913">QCodeListType IRIListType</td> </tr> <tr> <td data-bbox="842 913 1106 987">how (0..1) howuri (0..1)</td> <td data-bbox="1106 913 1444 987">QCodeType IRIType</td> </tr> <tr> <td data-bbox="842 987 1106 1061">why (0..1) whyuri (0..1)</td> <td data-bbox="1106 987 1444 1061">QCodeType IRIType</td> </tr> <tr> <td data-bbox="432 1061 842 1220" rowspan="3">▪ i18nAttributes (page 360)</td> <td data-bbox="842 1061 1106 1220"></td> <td data-bbox="1106 1061 1444 1220"></td> </tr> <tr> <td data-bbox="842 1106 1106 1151">xml:lang (0..1)</td> <td data-bbox="1106 1106 1444 1151">XML Schema language</td> </tr> <tr> <td data-bbox="842 1151 1106 1220">dir (0..1)</td> <td data-bbox="1106 1151 1444 1220">XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>		Name	Datatype	▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType	▪ i18nAttributes (page 360)			xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
		Name	Datatype																							
	▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID																							
		creator (0..1) creatoruri (0..1)	QCodeType IRIType																							
		modified (0..1)	DateOptTimeType																							
		custom (0..1)	XML Schema boolean																							
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType																							
		how (0..1) howuri (0..1)	QCodeType IRIType																							
		why (0..1) whyuri (0..1)	QCodeType IRIType																							
		▪ i18nAttributes (page 360)																								
xml:lang (0..1)			XML Schema language																							
dir (0..1)			XML Schema string: enumeration <i>ltr, rtl</i> .																							
Child Element(s)	<ul style="list-style-type: none"> ▪ groupRef (page 177) (0..unbounded) ▪ itemRef (page 206) (0..unbounded) ▪ conceptRef (page 102) (0..unbounded) ▪ title {itemMeta} (page 208) (0..unbounded) ▪ signal (page 292) (0..unbounded) ▪ edNote (page 153) (0..unbounded) ▪ groupExtProperty (page 176) (0..unbounded) 																									
XML Schema Note(s)	The local identifier (id) common to all elements at PCL provides a local identifier for groups.																									
Example(s)																										



14.6.101 Group Extension Property

Table 115. Group Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	groupExtProperty
Title	Group Extension Property
Definition	A generic extension property for a package group; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.102 Group Reference

Table 116. 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)	



14.6.103 Group Set

Table 117. 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)	<ul style="list-style-type: none"> ▪ root (1); XML Schema idref; The reference to a local group acting as the root of the hierarchy of groups.
Child Element(s)	<ul style="list-style-type: none"> ▪ group (page 174) (1..unbounded)
XML Schema Note(s)	
Example(s)	



14.6.104 Hash Value

Table 118. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
creator (0..1) creatoruri (0..1)		QCodeType IRIType	
modified (0..1)		DateOptTimeType	
custom (0..1)		XML Schema boolean	
pubconstraint (0..1) pubconstrainturi (0..1)		QCodeListType IRIListType	
how (0..1) howuri (0..1)		QCodeType IRIType	
why (0..1) whyuri (0..1)		QCodeType IRIType	
	<ul style="list-style-type: none"> ▪ hashtype (0..1); QCodeType (page 351); hashtypeuri (0..1); IRI Type (page 344); The hash algorithm used for creating the hash value The use of either the hashtype or the hashtypeuri attribute is mandatory (from NewsML-G2 v 2.19 onwards) ▪ scope (0..1); QCodeType (page 351); scopeuri (0..1); IRI Type (page 344); 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)			



14.6.105 Has Financial Instrument

Table 119. Has Financial Instrument

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	hasInstrument		
Title	Has Financial Instrument		
Definition	Defines a financial instrument which is related to a company.		
User Note(s)			
Implementation Note(s)	<p>The symbolsrc and symbol attributes are a pair of values which define the authority which issued a symbol and the issued symbol.</p> <p>The market can be defined in two ways: either by the market attribute which requires an identifier from a controlled vocabulary; or by a pair of marketlabelsrc and marketlabel values which define the authority which issued the marketlabel and the issued marketlabel.</p>		
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
	▪ symbol (1); XML Schema string; A symbol for the financial instrument		
	▪ symbolsrc (0..1); QCodeType; symbolsrcuri (0..1); IRIType; The source of the financial instrument symbol		
▪ market (0..1); QCodeType; marketuri (0..1); IRIType; A venue in which this financial instrument is traded			
▪ marketlabel (0..1); XML Scheme string; The label used for the market			
▪ marketlabelsrc (0..1); QCodeType; marketlabelsrcuri (0..1); IRIType; The source of the market label			
▪ type (0..1); QCodeListType; typeuri (0..1); IRIListType; The type(s) of the financial instrument			
▪ rank (0..1); XML Schema NonNegativeInteger; Indicates the relative importance among financial instruments of the same type. The same processing as for rank in the Ranking Attributes Group applies			
Child Element(s)			



Table 119. Has Financial Instrument

XML Schema Note(s)	
Example(s)	<pre> <hasInstrument symbol="RIO" symbolsrc="symsrc:MDNA" market="mic:XLON" marketlabel="LSE" marketlabelsrc="mlsrc:MDNA" type="instrtype:share" /> <hasInstrument symbol="RIO.L" symbolsrc="mic:RTSL" market="mic:XLON" marketlabel="LSE" marketlabelsrc="mlsrc:YahooFin" /> </pre>



14.6.106 **Headline**

Table 120. *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 345)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ rankingAttributes (page 362)	Name	Datatype
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.107 Hierarchy Info

Table 121. 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)	<p>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></p>



14.6.108 Hop

Table 122. 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)	<ul style="list-style-type: none"> ▪ 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)	<ul style="list-style-type: none"> ▪ party (page 247) (0..1) ▪ action (page 70) (0..unbounded)
XML Schema Note(s)	
Example(s)	



14.6.109 Hop History

Table 123. 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 184) (1..unbounded)
XML Schema Note(s)	
Example(s)	

**14.6.110 Icon**

Table 124. 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	

Table 124. Icon

Attribute(s)	<ul style="list-style-type: none"> ▪ rendition (0..1); QCodeType (page 351); renditionuri (0..1); IRI Type (page 344); 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. 		
	<ul style="list-style-type: none"> ▪ width (0..1); xsd:nonNegativeInteger; The width of of visual content. 		
	<ul style="list-style-type: none"> ▪ widthunit (0..1); QCodeType (page 351); widthunituri (0..1); IRIType (page 344); If present it defines the width unit for the width. 		
	<ul style="list-style-type: none"> ▪ height (0..1); xsd:nonNegativeInteger; The height of visual content. 		
	<ul style="list-style-type: none"> ▪ heightunit (0..1); QCodeType (page 351); heightunituri (0..1); IRI Type (page 344); If present it defines the height unit for the height. 		
	<ul style="list-style-type: none"> ▪ orientation 0..1; xs: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 Orientation Values (page 368) 		
	<ul style="list-style-type: none"> ▪ colourpace 0..1; QCodeType; colourpaceuri, 0..1; IRIType The colour space of an image. Applies to image icons. 		
	<ul style="list-style-type: none"> ▪ colourindicator 0..1; QCodeType; colourindicatoruri 0..1; IRIType; 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/ 		
	<ul style="list-style-type: none"> ▪ videocodec 0..1; QCodeType; videocodecuri 0..1; IRIType; The applicable codec for video data. Applies to video icons. The IPTC Video Codec NewsCodes may be used: http://cv.iptc.org/newscodes/videocodec/ 		
	<ul style="list-style-type: none"> ▪ targetResourceAttributes (page 365) 	Name	Datatype
		href (0..1)	IRIType
		residref (0..1)	XML Schema string
		version (0..1)	XML Schema positiveInteger
		contenttype (0..1)	XML Schema string
contenttypevariant (0..1)		XML Schema string	
format (0..1) formaturi (0..1)		QCodeType IRIType	
size (0..1)		XML Schema non NegativeInteger	
title (0..1)		XML Schema string	
<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype	
	id (0..1)	XML Schema ID	
	creator (0..1) creatoruri (0..1)	QCodeType IRIType	
	modified (0..1)	DateOptTimeType	
	custom (0..1)	XML Schema boolean	
	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	
how (0..1) howuri (0..1)	QCodeType IRIType		



Table 124. Icon

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

14.6.111 Incoming Feed Identifier

Table 125. Incoming Feed Identifier

(XML) Data Model	Element
Namespace (prefix)	nar
Name	incomingFeedId
Title	<i>Incoming Feed Identifier</i>
Definition	The identifier of an incoming feed. A feed identifier may be defined by i/ the provider of the feed and/or ii/ the processor of the feed.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	QCodePropType (page 350)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType (page 351); A refinement of the semantics of the property - expressed by a QCode ▪ roleuri (0..1); IRIType (page 344); A refinement of the semantics of the property - expressed by a URI
Child Element(s)	
XML Schema Note(s)	
Example(s)	<p>A Feed Identifier (PRNUKReg) supplied in the feed from a Provider (PRN in this case): <code><incomingFeedId role="feedIdRole:srcFeedId" qcode="PRNFeedId:PRNUKReg" /></code></p> <p>A Feed Identifier (RRR) assigned by the processor on receipt of the feed (by TR in this case): <code><incomingFeedId role="feedIdRole:TRFeedId" qcode="TRFeedId:RRR" /></code></p>

14.6.112 Inline Concept Marker

Table 126. 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)	<p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Mixed Content
Internally Ctrl Values	
Externally Ctrl Values	

Table 126. Inline Concept Marker (Continued)

	<ul style="list-style-type: none"> ▪ class (0..1); XML Schema String; An equivalent of the HTML <i>class</i> attribute. ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p>																		
	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as controlled or uncontrolled property value. 																		
Attribute(s)	<table border="1"> <thead> <tr> <th></th> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td rowspan="2">▪ i18nAttributes (page 360)</td> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>		Name	Datatype	▪ i18nAttributes (page 360)	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .										
		Name	Datatype																
	▪ i18nAttributes (page 360)	xml:lang (0..1)	XML Schema language																
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .																
	<table border="1"> <thead> <tr> <th></th> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td rowspan="7">▪ commonPowerAttributes (page 361)</td> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>		Name	Datatype	▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype																
	▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID																
		creator (0..1) creatoruri (0..1)	QCodeType IRIType																
		modified (0..1)	DateOptTimeType																
		custom (0..1)	XML Schema boolean																
pubconstraint (0..1) pubconstrainturi (0..1)		QCodeListType IRIListType																	
how (0..1) howuri (0..1)		QCodeType IRIType																	
why (0..1) whyuri (0..1)		QCodeType IRIType																	
<table border="1"> <thead> <tr> <th></th> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td rowspan="5">▪ quantifyAttributes (page 363)</td> <td>confidence (0..1)</td> <td>Int100Type</td> </tr> <tr> <td>relevance (0..1)</td> <td>Int100Type</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>derivedfrom (0..1) DEPRECATED</td> <td>QCodeType</td> </tr> </tbody> </table>		Name	Datatype	▪ quantifyAttributes (page 363)	confidence (0..1)	Int100Type	relevance (0..1)	Int100Type	why (0..1) whyuri (0..1)	QCodeType IRIType	derivedfrom (0..1) DEPRECATED	QCodeType							
	Name	Datatype																	
▪ quantifyAttributes (page 363)	confidence (0..1)	Int100Type																	
	relevance (0..1)	Int100Type																	
	why (0..1) whyuri (0..1)	QCodeType IRIType																	
	derivedfrom (0..1) DEPRECATED	QCodeType																	
		▪ span (page 294) (0..unbounded)																	
	▪ ruby (page 280) (0..unbounded)																		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.																		
XML Schema Note(s)																			
Example(s)																			

**14.6.113 Inline Data**Table 127. *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 127. Inline Data

	<ul style="list-style-type: none"> ▪ encoding (0..1); QCodeType (page 351); encodinguri (0..1); IRI Type (page 344); 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; formaturi (0..1); IRI Type (page 344); Refinement of a generic content type (i.e. IANA MIME type). 																																																																								
	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>rendition (0..1)</td> <td>QCodeType</td> </tr> <tr> <td>renditionuri (0..1)</td> <td>IRIType</td> </tr> <tr> <td>generator (0..1)</td> <td>XML Schema string</td> </tr> <tr> <td>generated (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>hascontent (0..1)</td> <td>XML Schema boolean</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	rendition (0..1)	QCodeType	renditionuri (0..1)	IRIType	generator (0..1)	XML Schema string	generated (0..1)	DateOptTimeType	hascontent (0..1)	XML Schema boolean																																																										
	Name	Datatype																																																																							
id (0..1)	XML Schema ID																																																																								
rendition (0..1)	QCodeType																																																																								
renditionuri (0..1)	IRIType																																																																								
generator (0..1)	XML Schema string																																																																								
generated (0..1)	DateOptTimeType																																																																								
hascontent (0..1)	XML Schema boolean																																																																								
Attribute(s)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>wordcount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>linecount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>pagecount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>width</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>widthunit</td> <td>QCodeType</td> </tr> <tr> <td>widthunituri</td> <td>IRIType</td> </tr> <tr> <td>height</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>heightunit</td> <td>QCodeType</td> </tr> <tr> <td>heightunituri</td> <td>IRIType</td> </tr> <tr> <td>orientation</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>layoutorientation</td> <td>QCodeType</td> </tr> <tr> <td>layoutorientationuri</td> <td>IRIType</td> </tr> <tr> <td>colourspace</td> <td>QCodeType</td> </tr> <tr> <td>colourspaceuri</td> <td>IRIType</td> </tr> <tr> <td>colourindicator</td> <td>QCodeType</td> </tr> <tr> <td>colourindicatoruri</td> <td>IRIType</td> </tr> <tr> <td>colourdepth</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>resolution</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>duration</td> <td>XML Schema string</td> </tr> <tr> <td>durationunit</td> <td>QCodeType</td> </tr> <tr> <td>durationunituri</td> <td>IRIType</td> </tr> <tr> <td>audiocodec</td> <td>QCodeType</td> </tr> <tr> <td>audiocodecuri</td> <td>IRIType</td> </tr> <tr> <td>audiobitrate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiovbr</td> <td>XML Schema boolean</td> </tr> <tr> <td>audiosamplesize</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiosamplerate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiochannels</td> <td>QCodeType</td> </tr> <tr> <td>audiochannelsuri</td> <td>IRIType</td> </tr> <tr> <td>videocodec</td> <td>QCodeType</td> </tr> <tr> <td>videocodecuri</td> <td>IRIType</td> </tr> <tr> <td>videoavgbitrate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>videovbr</td> <td>XML Schema boolean</td> </tr> <tr> <td>videoframerate</td> <td>XML Schema decimal</td> </tr> <tr> <td>videoscan</td> <td>enumeration progressive/interlaced</td> </tr> </tbody> </table>	Name	Datatype	wordcount	XML Schema nonNegativeInteger	linecount	XML Schema nonNegativeInteger	pagecount	XML Schema nonNegativeInteger	width	XML Schema nonNegativeInteger	widthunit	QCodeType	widthunituri	IRIType	height	XML Schema nonNegativeInteger	heightunit	QCodeType	heightunituri	IRIType	orientation	XML Schema nonNegativeInteger	layoutorientation	QCodeType	layoutorientationuri	IRIType	colourspace	QCodeType	colourspaceuri	IRIType	colourindicator	QCodeType	colourindicatoruri	IRIType	colourdepth	XML Schema nonNegativeInteger	resolution	XML Schema positiveInteger	duration	XML Schema string	durationunit	QCodeType	durationunituri	IRIType	audiocodec	QCodeType	audiocodecuri	IRIType	audiobitrate	XML Schema positiveInteger	audiovbr	XML Schema boolean	audiosamplesize	XML Schema positiveInteger	audiosamplerate	XML Schema positiveInteger	audiochannels	QCodeType	audiochannelsuri	IRIType	videocodec	QCodeType	videocodecuri	IRIType	videoavgbitrate	XML Schema positiveInteger	videovbr	XML Schema boolean	videoframerate	XML Schema decimal	videoscan	enumeration progressive/interlaced
Name	Datatype																																																																								
wordcount	XML Schema nonNegativeInteger																																																																								
linecount	XML Schema nonNegativeInteger																																																																								
pagecount	XML Schema nonNegativeInteger																																																																								
width	XML Schema nonNegativeInteger																																																																								
widthunit	QCodeType																																																																								
widthunituri	IRIType																																																																								
height	XML Schema nonNegativeInteger																																																																								
heightunit	QCodeType																																																																								
heightunituri	IRIType																																																																								
orientation	XML Schema nonNegativeInteger																																																																								
layoutorientation	QCodeType																																																																								
layoutorientationuri	IRIType																																																																								
colourspace	QCodeType																																																																								
colourspaceuri	IRIType																																																																								
colourindicator	QCodeType																																																																								
colourindicatoruri	IRIType																																																																								
colourdepth	XML Schema nonNegativeInteger																																																																								
resolution	XML Schema positiveInteger																																																																								
duration	XML Schema string																																																																								
durationunit	QCodeType																																																																								
durationunituri	IRIType																																																																								
audiocodec	QCodeType																																																																								
audiocodecuri	IRIType																																																																								
audiobitrate	XML Schema positiveInteger																																																																								
audiovbr	XML Schema boolean																																																																								
audiosamplesize	XML Schema positiveInteger																																																																								
audiosamplerate	XML Schema positiveInteger																																																																								
audiochannels	QCodeType																																																																								
audiochannelsuri	IRIType																																																																								
videocodec	QCodeType																																																																								
videocodecuri	IRIType																																																																								
videoavgbitrate	XML Schema positiveInteger																																																																								
videovbr	XML Schema boolean																																																																								
videoframerate	XML Schema decimal																																																																								
videoscan	enumeration progressive/interlaced																																																																								



Table 127. Inline Data

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



14.6.114 Inline Reference

Table 128. *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 326)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ idrefs (0..1); XML Schema IDREFS; A set of local identifiers of inline content. 		
	<ul style="list-style-type: none"> ▪ quantifyAttributes (page 363) 	Name	Datatype
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
		why (0..1) whyuri (0..1)	QCodeType IRIType
derivedfrom (0..1) DEPRECATED	QCodeType		
Child Element(s)			
XML Schema Note(s)			
Example(s)			

**14.6.115 Inline XML**Table 129. *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 129. Inline XML

	<ul style="list-style-type: none"> ▪ contenttype (0..1); XML Schema string; The IANA (Internet Assigned Numbers Authority) MIME type of the target resource. 																																																					
	<ul style="list-style-type: none"> ▪ format (0..1); QCodeType; formaturi (0..1); IRI Type (page 344); Refinement of a generic content type (i.e. IANA MIME type). 																																																					
Attribute(s)	<ul style="list-style-type: none"> ▪ newsContentAttributes (page 364) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>rendition (0..1) renditionuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>generator (0..1)</td> <td>XML Schema string</td> </tr> <tr> <td>generated (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>hascontent (0..1)</td> <td>XML Schema boolean</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	rendition (0..1) renditionuri (0..1)	QCodeType IRIType	generator (0..1)	XML Schema string	generated (0..1)	DateOptTimeType	hascontent (0..1)	XML Schema boolean																																								
		Name	Datatype																																																			
		id (0..1)	XML Schema ID																																																			
		rendition (0..1) renditionuri (0..1)	QCodeType IRIType																																																			
		generator (0..1)	XML Schema string																																																			
		generated (0..1)	DateOptTimeType																																																			
	hascontent (0..1)	XML Schema boolean																																																				
	<ul style="list-style-type: none"> ▪ newsContentCharacteristics (page 365) (all: 0..1) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>wordcount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>linecount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>pagecount</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>width</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>widthunit widthunituri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>height</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>heightunit heightunituri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>orientation</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>layoutorientation layoutorientationuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>colourspace colourspaceuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>colourindicator colourindicatoruri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>colourdepth</td> <td>XML Schema nonNegativeInteger</td> </tr> <tr> <td>resolution</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>duration</td> <td>XML Schema string</td> </tr> <tr> <td>durationunit durationunituri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>audiocodec audiocodecuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>audiobitrate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiovbr</td> <td>XML Schema boolean</td> </tr> <tr> <td>audiosamplesize</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiosamplerate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>audiochannels audiochannelsuri</td> <td>QCodeType audiochannelsuri</td> </tr> <tr> <td>videocodec videocodecuri</td> <td>QCodeType IRIType</td> </tr> <tr> <td>videoavgbitrate</td> <td>XML Schema positiveInteger</td> </tr> <tr> <td>videovbr</td> <td>XML Schema boolean</td> </tr> <tr> <td>videoframerate</td> <td>XML Schema decimal</td> </tr> </tbody> </table>	Name	Datatype	wordcount	XML Schema nonNegativeInteger	linecount	XML Schema nonNegativeInteger	pagecount	XML Schema nonNegativeInteger	width	XML Schema nonNegativeInteger	widthunit widthunituri	QCodeType IRIType	height	XML Schema nonNegativeInteger	heightunit heightunituri	QCodeType IRIType	orientation	XML Schema nonNegativeInteger	layoutorientation layoutorientationuri	QCodeType IRIType	colourspace colourspaceuri	QCodeType IRIType	colourindicator colourindicatoruri	QCodeType IRIType	colourdepth	XML Schema nonNegativeInteger	resolution	XML Schema positiveInteger	duration	XML Schema string	durationunit durationunituri	QCodeType IRIType	audiocodec audiocodecuri	QCodeType IRIType	audiobitrate	XML Schema positiveInteger	audiovbr	XML Schema boolean	audiosamplesize	XML Schema positiveInteger	audiosamplerate	XML Schema positiveInteger	audiochannels audiochannelsuri	QCodeType audiochannelsuri	videocodec videocodecuri	QCodeType IRIType	videoavgbitrate	XML Schema positiveInteger	videovbr	XML Schema boolean	videoframerate	XML Schema decimal
		Name	Datatype																																																			
		wordcount	XML Schema nonNegativeInteger																																																			
		linecount	XML Schema nonNegativeInteger																																																			
		pagecount	XML Schema nonNegativeInteger																																																			
		width	XML Schema nonNegativeInteger																																																			
		widthunit widthunituri	QCodeType IRIType																																																			
		height	XML Schema nonNegativeInteger																																																			
heightunit heightunituri		QCodeType IRIType																																																				
orientation		XML Schema nonNegativeInteger																																																				
layoutorientation layoutorientationuri		QCodeType IRIType																																																				
colourspace colourspaceuri		QCodeType IRIType																																																				
colourindicator colourindicatoruri		QCodeType IRIType																																																				
colourdepth		XML Schema nonNegativeInteger																																																				
resolution		XML Schema positiveInteger																																																				
duration		XML Schema string																																																				
durationunit durationunituri		QCodeType IRIType																																																				
audiocodec audiocodecuri		QCodeType IRIType																																																				
audiobitrate		XML Schema positiveInteger																																																				
audiovbr	XML Schema boolean																																																					
audiosamplesize	XML Schema positiveInteger																																																					
audiosamplerate	XML Schema positiveInteger																																																					
audiochannels audiochannelsuri	QCodeType audiochannelsuri																																																					
videocodec videocodecuri	QCodeType IRIType																																																					
videoavgbitrate	XML Schema positiveInteger																																																					
videovbr	XML Schema boolean																																																					
videoframerate	XML Schema decimal																																																					



Table 129. Inline XML

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

**14.6.116 Instance Of**

Table 130. 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 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.117 Instant Messaging Address

Table 131. 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 322)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.118 Information Source

Table 132. Information Source

(XML) Data Model	Element
Namespace (prefix)	nar
Name	infoSource
Title	Information Source
Definition	A party (person or organisation) which originated, distributed, aggregated or supplied the content or provided some information used to create or enhance the content.
User Note(s)	<p>If no @role is applied the information source provided some information used to create or enhance the content and played no other role. Omitting @role is equivalent to applying http://cv.iptc.org/newscodes/infosourcerole/origininfo as the only role value.</p> <p>If a party did anything other than originate information a role attribute with one or more roles must be applied. The recommended vocabulary is the IPTC Information Source Roles NewsCodes at http://cv.iptc.org/newscodes/infosourcerole/</p> <p>To indicate that a party has modified or enhanced the content use the contributor property.</p> <p>If an entity plays more than one role, the <i>infoSource</i> element has to be included multiple times, with different values of role.</p>
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flex1PartyPropType (page 324)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the <i>role</i> attribute: http://cv.iptc.org/newscodes/infosourcerole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.119 Item Class

Table 133. 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 <i>concept</i> . 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 352)
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)	



14.6.120 Item Count

Table 134. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
	why (0..1) whyuri (0..1)	QCodeType IRIType	
		<ul style="list-style-type: none"> ▪ rangefrom (1); XML Schema nonNegativeInteger; The lower limit of the range of planned items 	
		<ul style="list-style-type: none"> ▪ rangeto (1); XML Schema positiveInteger; The upper limit of the range of planned items 	
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.121 Item Metadata

Table 135. 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)	<ul style="list-style-type: none"> ▪ Item Management Group (page 309) (1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>provider (1)</td> <td>118</td> </tr> <tr> <td>embargoed (0..1)</td> <td>131</td> </tr> <tr> <td>firstCreated (0..1)</td> <td>132</td> </tr> <tr> <td>versionCreated (1)</td> <td>133</td> </tr> <tr> <td>edNote (0..unbounded)</td> <td>153</td> </tr> <tr> <td>service (0..unbounded)</td> <td>154</td> </tr> <tr> <td>filename (0..1)</td> <td>167</td> </tr> <tr> <td>itemClass (1)</td> <td>201</td> </tr> <tr> <td>pubStatus (0..1)</td> <td>261</td> </tr> <tr> <td>role (0..1)</td> <td>279</td> </tr> <tr> <td>title {itemMeta} (page 208) (0..unbounded)</td> <td></td> </tr> <tr> <td>altRep (0..unbounded)</td> <td>75</td> </tr> <tr> <td>generator (0..unbounded)</td> <td>170</td> </tr> <tr> <td>instanceOf (0..unbounded)</td> <td>198</td> </tr> <tr> <td>memberOf (0..unbounded)</td> <td>219</td> </tr> <tr> <td>profile (0..1)</td> <td>259</td> </tr> <tr> <td>signal (0..unbounded)</td> <td>292</td> </tr> <tr> <td>deliverableOf (0..unbounded)</td> <td>147</td> </tr> <tr> <td>hash (0..unbounded)</td> <td>179</td> </tr> <tr> <td>expires (0..unbounded)</td> <td>164</td> </tr> </tbody> </table>	Element Name	Page	provider (1)	118	embargoed (0..1)	131	firstCreated (0..1)	132	versionCreated (1)	133	edNote (0..unbounded)	153	service (0..unbounded)	154	filename (0..1)	167	itemClass (1)	201	pubStatus (0..1)	261	role (0..1)	279	title {itemMeta} (page 208) (0..unbounded)		altRep (0..unbounded)	75	generator (0..unbounded)	170	instanceOf (0..unbounded)	198	memberOf (0..unbounded)	219	profile (0..1)	259	signal (0..unbounded)	292	deliverableOf (0..unbounded)	147	hash (0..unbounded)	179	expires (0..unbounded)	164
		Element Name	Page																																									
		provider (1)	118																																									
		embargoed (0..1)	131																																									
		firstCreated (0..1)	132																																									
		versionCreated (1)	133																																									
		edNote (0..unbounded)	153																																									
		service (0..unbounded)	154																																									
		filename (0..1)	167																																									
		itemClass (1)	201																																									
		pubStatus (0..1)	261																																									
		role (0..1)	279																																									
		title {itemMeta} (page 208) (0..unbounded)																																										
		altRep (0..unbounded)	75																																									
		generator (0..unbounded)	170																																									
		instanceOf (0..unbounded)	198																																									
		memberOf (0..unbounded)	219																																									
		profile (0..1)	259																																									
		signal (0..unbounded)	292																																									
		deliverableOf (0..unbounded)	147																																									
hash (0..unbounded)	179																																											
expires (0..unbounded)	164																																											
▪ link (page 214) (0..unbounded)																																												
▪ itemMetaExtProperty (page 205) (0..unbounded)																																												
▪ Extension Point (0..unbounded). Any set of provider-defined properties.																																												



Table 135. *Item Metadata (Continued)*

XML Schema Note(s)	
<p>Example(s)</p>	<p>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)</p> <pre data-bbox="438 376 1428 533"><itemMeta> ... <hash hashtype="htype:MD5" scope="hscope:provmix">hash-value..... </hash> </itemMeta></pre>



14.6.122 Item Metadata Extension Property

Table 136. Item Metadata Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	itemMetaExtProperty
Title	Item Metadata Extension Property
Definition	A generic extension property for item metadata; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.123 Item Reference

Table 137. 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 346)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.124 Item Set

Table 138. 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)	<ul style="list-style-type: none"> ▪ newsItem (page 232) (1..unbounded) ▪ conceptItem (page 99) (1..unbounded) ▪ packageItem (page 241) (1..unbounded) ▪ knowledgeItem (page 210) (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)	

**14.6.125 Item Title**Table 139. *Item Title*

(XML) Data Model	Element
Namespace (prefix)	nar
Name	title {itemMeta}
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 345)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.126 Keyword

Table 140. 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 Normalized String type - see XML Schema specifications.		
XML Schema Spec	At: PCL		
Datatype	Extends Int1StringType (page 342)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the keyword. ▪ confidence, 0..1; QCodeType (page 351); The confidence with which the metadata has been assigned. ▪ relevance, 0..1; QCodeType (page 351); The relevance of the metadata to the news content to which it was attached. 		
	▪ quantifyAttributes (page 363)	Name	Datatype
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
	▪ rankingAttributes (page 362)	Name	Datatype
		why (0..1) whyuri (0..1) rank (0..1)	QCodeType IRIType XML Schema nonNegativeIntege
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.127 Knowledge Item

Table 141. Knowledge Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	knowledgeltem
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 109) (0..1)
	▪ partMeta (page 244) (0..unbounded)
	▪ assert (page 77) (0..unbounded)
	▪ inlineRef (page 194) (0..unbounded)
	▪ derivedFrom (page 150) (0..unbounded)
	▪ conceptSet (page 103) (0..1)
▪ schemeMeta (page 289) (0..1)	
XML Schema Note(s)	
Example(s)	



14.6.128 Language

Table 142. 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	<p>tag values must be valid BCP 47 language tags. Recommended IPTC NewsCodes for the <i>role</i> attribute: http://cv.iptc.org/newscodes/languagerole/</p>		
Attribute(s)	<ul style="list-style-type: none"> ▪ tag (1); XML Schema language; Indicator of the language. ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the property. 		
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
	<ul style="list-style-type: none"> ▪ rankingAttributes (page 362) 	Name	Datatype
rank (0..1)		XML Schema nonNegativeInteger	
Child Element(s)	<ul style="list-style-type: none"> ▪ name (page 260) (0..unbounded) 		
XML Schema Note(s)			
Example(s)			



14.6.129 Line

Table 143. Line

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	line {geoArea}		
Title	Line as geoArea		
Definition	Defines a line as a geographic area by listing two or more points.		
User Note(s)			
Implementation Note(s)	Order of positions has to be considered, a minimum of two position elements is mandatory		
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)	▪ position (page 172) (1..unbounded)		
XML Schema Note(s)			
Example(s)	<pre> <geoAreaDetails> <line> <position ... > <position ...> </line> </geoAreaDetail> </pre>		

**14.6.130 Line Break**

Table 144. 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)	

14.6.131 Link

Table 145. 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)	<p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p>
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Link1Type (page 346)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

**14.6.132 Locality**Table 146. *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 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType; roleuri (0..1); IRIType; Refines the semantics of the property
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.133 Located

Table 147. Located

(XML) Data Model	Element
Namespace (prefix)	nar
Name	located
Title	Located
Definition	A location from which the content originates.
User Note(s)	<p>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:</p> <ul style="list-style-type: none"> - 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 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.134 Location

Table 148. 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 328)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ timeValidityAttributes (page 361) 	Name	DataType
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.135 Location Details

Table 149. 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 313)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

**14.6.136 Member Of**

Table 150. 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 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.137 Message Header

Table 151. 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)	<ul style="list-style-type: none"> ▪ sent (page 142) (1); May not be updated in case of a message retransmission. ▪ sender (page 291) (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 270) (0..unbounded) ▪ catalog (page 86) (0..unbounded) ▪ transmitId (page 300) (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 258) (0..1) ▪ origin (page 237) (0..1); This string structure is not specified by the IPTC. ▪ destination (page 151) (0..unbounded) ▪ channel {News Message} (page 92) (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 299) (0..unbounded) ▪ signal (page 292) (0..unbounded) ▪ headerExtProperty (page 221) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	



14.6.138 Message Header Extension Property

Table 152. Message Header Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	headerExtProperty
Title	Message Header Extension Property
Definition	A generic extension property for a message header; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

**14.6.139 Narrower**

Table 153. 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 354)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.140 News Content Characteristics

Table 154. News Content Characteristics

(XML) Data Model	Element
Namespace (prefix)	nar
Name	newsContentCharacteristics
Title	News Content Characteristics
Definition	The characteristics of the content of a News Item.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	Mandatory IPTC NewsCodes: http://cv.iptc.org/newscodes/dimensionunit/



Table 154. News Content Characteristics

Attribute(s)	Name	Datatype
	id (0..1)	XML Schema ID
	creator (0..1) creatoruri (0..1)	QCodeType IRIType
	modified (0..1)	DateOptTimeType
	custom (0..1)	XML Schema boolean
	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
	how (0..1) howuri (0..1)	QCodeType IRIType
	why (0..1) whyuri (0..1)	QCodeType IRIType
	wordcount	XML Schema nonNegativeInteger
	linecount	XML Schema nonNegativeInteger
	pagecount	XML Schema nonNegativeInteger
	width	XML Schema nonNegativeInteger
	widthunit widthunituri	QCodeType IRIType
	height	XML Schema nonNegativeInteger
	heightunit heightunituri	QCodeType IRIType
	orientation	XML S nonNegativeInteger
	layoutorientation layoutorientationuri	QCodeType IRIType
	colourspace colourspaceuri	QCodeType IRIType
	colourindicator colourindicatoruri	QCodeType IRIType
	colourdepth	XML Schema nonNegativeInteger
	resolution	XML Schema positiveInteger
	duration	XML Schema string
	durationunit durationunituri	QCodeType IRIType
	audiocodec audiocodecuri	QCodeType IRIType
	audiobitrate	XML Schema positiveInteger
	audiovbr	XML Schema boolean
	audiosamplesize	XML Schema positiveInteger
	audiosamplerate	XML Schema positiveInteger
	audiochannels audiochannelsuri	QCodeType IRIType
	videocodec videocodecuri	QCodeType IRIType
	videoavgbitrate	XML Schema positiveInteger
	videovbr	XML Schema boolean

▪ **commonPowerAttributes** (page 361)

▪ **newsContentCharacteristics** (page 365)
(all: 0..1)



Table 154. News Content Characteristics

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



14.6.141 News Coverage (Concept Item)

Table 155. 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)	<ul style="list-style-type: none"> role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); Refines the semantics of the property. 																	
	<ul style="list-style-type: none"> commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	



Table 155. News Coverage for a Concept Item (Continued)

Child Element(s)	▪ g2ContentType (page 168) (0..1)		
	▪ itemClass (page 169) (0..1)		
	▪ assignedTo (page 79) (0..1)		
	▪ scheduled (page 287) (0..1)		
	▪ service (page 154) (0..unbounded)		
	▪ edNote (page 153) (0..unbounded)		
		Element Name	Page
		by (0..unbounded)	84
		creditline (0..unbounded)	126
		dateline (0..unbounded)	143
		description (0..unbounded)	146
		▪ Descriptive Metadata Group (page 309) (0..1)	
	genre (0..unbounded)	171	
	headline (0..unbounded)	182	
	keyword (0..unbounded)	209	
	language (0..unbounded)	211	
	slugline (0..unbounded)	293	
	subject (0..unbounded)	296	
XML Schema Note(s)			
Example(s)			



14.6.142 News Coverage (Planning Item)

Table 156. 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: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPower Attributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)	<ul style="list-style-type: none"> ▪ planning (page 250) (1) ▪ delivery (page 149) (0..1) ▪ newsCoverageExtProperty (page 229) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 		
	XML Schema Note(s)		
	Example(s)		



14.6.143 News Coverage Extension Property

Table 157. News Coverage Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	newsCoverageExtProperty
Title	News Coverage Extension Property
Definition	A generic extension property for news coverage; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.144 News Coverage Set

Table 158. 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 228) (1..unbounded)
XML Schema Note(s)	
Example(s)	



14.6.145 News Coverage Status

Table 159. 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 352)
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)	



14.6.146 News Item

Table 160. 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 310)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ contentMeta {News} (page 111) (0..1)
	▪ partMeta (page 244) (0..1)
	▪ assert (page 77) (0..unbounded)
	▪ inlineRef (page 194) (0..unbounded)
	▪ derivedFrom (page 150) (0..unbounded)
	▪ contentSet (page 119) (0..1)
XML Schema Note(s)	
Example(s)	



14.6.147 News Message

Table 161. 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)	<ul style="list-style-type: none"> ▪ header (page 220) (1) ▪ itemSet (page 207) (1)
XML Schema Note(s)	
Example(s)	



14.6.148 Object Details

Table 162. 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)	<ul style="list-style-type: none"> ▪ created {Object} (page 135) (0..1) ▪ creator (page 125) (0..unbounded) ▪ copyrightNotice (page 122) (0..unbounded) ▪ ceasedToExist {Object} (page 127) (0..1) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	



14.6.149 Occurrence Status

Table 163. 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 350)
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)	



14.6.150 Opening Hours

Table 164. 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 345)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.151 Origin

Table 165. 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. Rule for @qcode and @uri in an element: - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence.		
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 363)	Name	Datatype
		qcode (0..1)	QCodeType
		uri (0..1)	XML Schema anyURI
		literal (0..1)	NormalizedStringType (of G2)
		type typeuri	QCodeType IRIType
		role roleuri	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.152 Organisation Details

Table 166. 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: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ founded {Organisation} (page 141) (0..1)
	▪ dissolved {Organisation} (page 139) (0..1)
	▪ location (page 217) (0..unbounded)
	▪ contactInfo (page 105) (0..unbounded)
	▪ hasInstrument (page 180) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

**14.6.153 Organiser**

Table 167. 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 324)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the <i>role</i> attribute: http://cv.iptc.org/newscodes/eventorganiserrole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.154 Original Representation

Table 168. Original Representation

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	origRep		
Title	Original Representation		
Definition	An IRI which, upon dereferencing, provides the original representation of the Item, the IRI should be persistent.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	IRIType (page 344)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ timeValidityAttributes (page 361) 	Name	DataType
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
	<ul style="list-style-type: none"> ▪ accesstype (0..1); QCodeType; A qualifier which indicates the technical variant for accessing this item (e.g. communication protocols) - expressed by a QCode. ▪ accesstypeuri (0..1); IRIType; A qualifier which indicates the technical variant for accessing this item (e.g. communication protocols) - expressed by a URI 		
	<ul style="list-style-type: none"> ▪ reposrole (0..1); QCodeType/IRIType; A qualifier which indicates the role of the stated repository - expressed by a QCode ▪ reposroleuri (0..1); QCodeType/IRIType; A qualifier which indicates the role of the stated repository - expressed by a URI 		
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.155 Package Item

Table 169. 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	AnylItemType (page 310)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ contentMeta {Package} (page 113) (0..1)
	▪ partMeta (page 244) (0..unbounded)
	▪ assert (page 77) (0..unbounded)
	▪ inlineRef (page 194) (0..unbounded)
	▪ derivedFrom (page 150) (0..unbounded)
	▪ groupSet (page 178) (0..1)
XML Schema Note(s)	
Example(s)	

**14.6.156 Participant**

Table 170. 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 324)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for <i>role</i> attribute: http://cv.iptc.org/newscodes/eventparticipantrole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.157 Participation Requirement

Table 171. 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 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); Refines the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.158 Part of Content Metadata

Table 172. Part of Content Metadata

(XML) Data Model	Element							
Namespace (prefix)	nar							
Name	partMeta							
Title	Part of Content Metadata							
Definition	<p>A set of properties associated with a specific part of the content of the Item.</p> <p>The relationship of properties inside this partMeta and properties at a higher hierarchical level of the content parts structure is:</p> <ul style="list-style-type: none"> ▪ the semantic assertion of all properties at a higher level is inherited by this partMeta element as if these properties would be its children ▪ a child property of a specific name wipes out for this partMeta element any semantic assertions of properties of the same name at higher levels ▪ in this latter case: if the semantic assertion of a property at a higher level should be reinstated for this part of the content then this property has to appear again as child of this partMeta 							
User Note(s)								
Implementation Note(s)								
XML Schema Spec	At: PCL							
Datatype								
Internally Ctrl Values								
Externally Ctrl Values								
Attribute(s)	<ul style="list-style-type: none"> ▪ 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 360)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr</i>, <i>rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .
		Name	Datatype					
		xml:lang (0..1)	XML Schema language					
dir (0..1)	XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .							



Table 172. Part of Content Metadata (Continued)

	<ul style="list-style-type: none"> ▪ role (page 278) (0..1) ▪ icon (page 186) (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 297) (0..1) ▪ regionDelim (page 305) (0..1) 																										
Child Element(s)	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>urgency (0..1)</td> <td>303</td> </tr> <tr> <td>contentCreated (0..1)</td> <td>129</td> </tr> <tr> <td>contentModified (0..1)</td> <td>130</td> </tr> <tr> <td>located (0..unbounded)</td> <td>216</td> </tr> <tr> <td>infoSource (0..unbounded)</td> <td>200</td> </tr> <tr> <td>creator (0..unbounded)</td> <td>125</td> </tr> <tr> <td>contributor (0..unbounded)</td> <td>120</td> </tr> <tr> <td>audience (0..unbounded)</td> <td>80</td> </tr> <tr> <td>exclAudience (0..unbounded)</td> <td>161</td> </tr> <tr> <td>altId (0..unbounded)</td> <td>73</td> </tr> <tr> <td>rating (page 262) (0..unbounded)</td> <td></td> </tr> <tr> <td>userInteraction (page 302) (0..unbounded)</td> <td></td> </tr> </tbody> </table>	Element Name	Page	urgency (0..1)	303	contentCreated (0..1)	129	contentModified (0..1)	130	located (0..unbounded)	216	infoSource (0..unbounded)	200	creator (0..unbounded)	125	contributor (0..unbounded)	120	audience (0..unbounded)	80	exclAudience (0..unbounded)	161	altId (0..unbounded)	73	rating (page 262) (0..unbounded)		userInteraction (page 302) (0..unbounded)	
	Element Name	Page																									
	urgency (0..1)	303																									
	contentCreated (0..1)	129																									
	contentModified (0..1)	130																									
	located (0..unbounded)	216																									
	infoSource (0..unbounded)	200																									
	creator (0..unbounded)	125																									
	contributor (0..unbounded)	120																									
	audience (0..unbounded)	80																									
	exclAudience (0..unbounded)	161																									
	altId (0..unbounded)	73																									
	rating (page 262) (0..unbounded)																										
	userInteraction (page 302) (0..unbounded)																										
	Child Element(s)	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>by (0..unbounded)</td> <td>84</td> </tr> <tr> <td>creditline (0..unbounded)</td> <td>126</td> </tr> <tr> <td>dateline (0..unbounded)</td> <td>143</td> </tr> <tr> <td>description (0..unbounded)</td> <td>146</td> </tr> <tr> <td>genre (0..unbounded)</td> <td>171</td> </tr> <tr> <td>headline (0..unbounded)</td> <td>182</td> </tr> <tr> <td>keyword (0..unbounded)</td> <td>209</td> </tr> <tr> <td>language (0..unbounded)</td> <td>211</td> </tr> <tr> <td>slugline (0..unbounded)</td> <td>293</td> </tr> <tr> <td>subject (0..unbounded)</td> <td>296</td> </tr> </tbody> </table>	Element Name	Page	by (0..unbounded)	84	creditline (0..unbounded)	126	dateline (0..unbounded)	143	description (0..unbounded)	146	genre (0..unbounded)	171	headline (0..unbounded)	182	keyword (0..unbounded)	209	language (0..unbounded)	211	slugline (0..unbounded)	293	subject (0..unbounded)	296			
Element Name		Page																									
by (0..unbounded)		84																									
creditline (0..unbounded)		126																									
dateline (0..unbounded)		143																									
description (0..unbounded)		146																									
genre (0..unbounded)		171																									
headline (0..unbounded)		182																									
keyword (0..unbounded)		209																									
language (0..unbounded)		211																									
slugline (0..unbounded)	293																										
subject (0..unbounded)	296																										
	<ul style="list-style-type: none"> ▪ partMetaExtProperty (page 246) (0..unbounded) ▪ signal (page 292) (0..unbounded) ▪ edNote (page 153) (0..unbounded) ▪ link (page 214) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 																										
XML Schema Note(s)																											
Example(s)	<pre><!-- 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></pre>																										



14.6.159 Part of Content Metadata Extension Property

Table 173. Part of Content Metadata Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	partMetaExtProperty
Title	Part of Content Metadata Extension Property
Definition	A generic extension property for part of content metadata; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

**14.6.160 Party (Hop History)**

Table 174. 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 324)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.161 Person Details

Table 175. 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 134) (0..1)
	▪ died (page 137) (0..1)
	▪ affiliation (page 72) (0..unbounded)
	▪ contactInfo (page 105) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	



14.6.162 Phone Number

Table 176. Phone Number

(XML) Data Model	Element
Namespace (prefix)	nar
Name	phone
Title	Phone Number
Definition	An international phone number.
User Note(s)	The <i>tech</i> attribute indicates a land-line, cellular etc., service.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	ElectronicAddressTechType (page 322)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.163 Planning Details

Table 177. 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)	<ul style="list-style-type: none"> ▪ ▪ 		
Child Element(s)	<ul style="list-style-type: none"> ▪ g2ContentType (page 168) (0..1) ▪ itemClass (page 169) (0..1) ▪ itemCount (page 202) (0..1) ▪ assignedTo (page 79) (0..1) ▪ scheduled (page 287) (0..1) ▪ service (page 154) (0..unbounded) ▪ edNote (page 153) (0..unbounded) ▪ newsContentCharacteristics (page 223) (0..unbounded) ▪ planningExtProperty (page 251) (0..unbounded) ▪ Extension Point (0..unbounded) Any set of provider-defined properties. 		
	<ul style="list-style-type: none"> ▪ Descriptive Metadata Group (page 309) (0..1) 	Element Name	Page
		by (0..unbounded)	84
		creditline (0..unbounded)	126
		dateline (0..unbounded)	143
		description (0..unbounded)	146
		genre (0..unbounded)	171
		headline (0..unbounded)	182
		keyword (0..unbounded)	209
		language (0..unbounded)	211
		slugline (0..unbounded)	293
	subject (0..unbounded)	296	
	XML Schema Note(s)		
Example(s)			



14.6.164 Planning Extension Property

Table 178. Planning Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	planningExtProperty
Title	Planning Extension Property
Definition	A generic extension property for planning; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.165 Planning Item

Table 179. 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	AnylItemType (page 310)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ contentMeta {Planning} (page 115) (0..1)
	▪ assert (page 77) (0..unbounded)
	▪ inlineRef (page 194) (0..unbounded)
	▪ derivedFrom (page 150) (0..unbounded)
	▪ newsCoverageSet (page 230) (0..1)
XML Schema Note(s)	
Example(s)	



14.6.166 POI Details

Table 180. 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)	<ul style="list-style-type: none"> ▪ position (page 172) (0..1) ▪ address {POI} (page 256) (0..1) ▪ openHours (page 236) (0..1) ▪ capacity (page 85) (0..1) ▪ access (page 67) (0..unbounded) ▪ details (page 218) (0..unbounded) ▪ contactInfo (page 105) (0..unbounded) ▪ created {POI} (page 136) (0..1) ▪ ceasedToExist {POI} (page 128) (0..1) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	<pre> <POIDetails> <address> <line role="arol:street">Friedrichstraße 22</line> <line role="arol:buildingname">Hofbräuhaus</line> <locality role="arol:sublocality"> <name>Schwabing</name> </locality> <locality role="arol:locality"> <name>München</name> </locality> <area role="arol:subarea"> <name>Oberbayern</name> </area> <area role="arol:area"> <name>Bayern</name> </area> <country qcode="iso3:DEU"/> </address> </POIDetails> </pre>



14.6.167 Polygon

Table 181. Polygon

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	polygon		
Title	Polygon as geoArea		
Definition	Defines a polygon as geographic area by a listing of three or more points.		
User Note(s)			
Implementation Note(s)	Order of positions has to be considered, a minimum of three position elements is mandatory		
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)	▪ position (page 172) (1..unbounded)		
XML Schema Note(s)			
Example(s)	<pre> <geoAreaDetails> <polygon> <position ... > <position ...> </polygon> </geoAreaDetail> </pre>		

14.6.168 Postal Address

Table 182. 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)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the postal address.
Child Element(s)	▪ line {address} (page 71) (0..unbounded)
	▪ locality (page 215) (0..unbounded)
	▪ area (page 124) (0..unbounded)
	▪ country (page 123) (0..1)
	▪ postalCode (page 257) (0..1)
XML Schema Note(s)	
Example(s)	

14.6.169 Postal Address of a Point of Interest

Table 183. Postal Address of A Point of Interest

(XML) Data Model	Element
Namespace (prefix)	nar
Name	address {POI}
Title	Postal Address 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 {address} (page 71) (0..unbounded)
	▪ locality (page 215) (0..1)
	▪ area (page 124) (0..1)
	▪ country (page 123) (0..1)
	▪ postalCode (page 257) (0..1)
XML Schema Note(s)	
Example(s)	



14.6.170 Postal Code

Table 184. 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 342)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

**14.6.171 Priority**

Table 185. 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 341)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

**14.6.172 Profile**

Table 186. 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 359)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.173 Property Value Name

Table 187. 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: PCL
Datatype	IntlStringType (page 342)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the postal address ▪ part (0..1); QCodeType (page 351); parturi (0..1); IRIType (page 344); Specifies which part of a full name this property provides ▪ validfrom (0..1); DateOptTimeType (page 318); The date (and, optionally, the time) before which a relationship is not valid. ▪ validto (0..1); DateOptTimeType (page 318); The date (and, optionally, the time) after which a relationship is not valid.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.174 Publish Status

Table 188. 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 352)
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)	



14.6.175 Rating

Table 189. Rating of Content

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	rating		
Title	Rating of Content		
Definition	Expresses the rating of the content of this item by a party..		
User Note(s)	<p>On @raters:</p> <p>a) If @raters is not present the number of raters defaults to 1</p> <p>b) @raters does not require that the count indicates distinct persons.</p>		
Implementation Note(s)	<p>on @valcalctype:</p> <p>A CV for the calculation type should include: mean, median, sum, unknown</p>		
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
	▪ value (1); XML Schema Decimal; The rating of the content expressed as decimal value from a scale.		
	▪ valcalctype (0..1); QCodeType; valcalctypeuri (0..1); IRIType; Indicates how the value was calculated (methods like median, average ...).		
▪ scalemin (1); XML Schema Decimal; The minimum value of the rating scale.			
▪ scalemax (1); XML Schema Decimal; The maximum value of the rating scale.			
▪ scaleunit (0..1); QCodeType; scaleunituri (0..1); IRIType; The units which apply to the rating scale. The use of either the scaleunit or the scaleunituri attribute is mandatory (from NewsML-G2 v 2.19 onwards)			
▪ raters (0..1); XML Schema NonNegativeInteger; The number of parties acting as raters.			
▪ ratertype (0..1); QCodeType; ratertypeuri (0..1); IRIType; The type of the rating parties.			
▪ ratingtype (0..1); QCodeType; ratingtypeuri (0..1); IRIType; Full definition of the rating.			



Table 189. Rating of Content

Child Element(s)	
XML Schema Note(s)	
Example(s)	<pre><rating value="2.3" valcalctype="rcalctype:amean" scalemin="1" scalemax="5" scaleunit="rscaleunit:imgstar" raters="57" rater-type="rtype:customer" /></pre>



14.6.176 Recurrence Date

Table 190. 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 317)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.177 Recurrence Rule

Table 191. 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 353)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.178 Registration

Table 192. 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 313)
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)	



14.6.179 Related Concept

Table 193. 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 354)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ value; (0..1); XML Schema String; The related property's value (see also: qcode, uri 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 351); valueunituri (0..1); IRI Type (page 344); The unit of the value attribute.
Child Element(s)	<ul style="list-style-type: none"> ▪ bag (page 81) (0..1)



Table 193. Related Concept

XML Schema Note(s)



Table 193. Related Concept

Note(s) & Example(s)	<p>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:</p> <table border="0"> <tr> <td>Subject</td> <td>Predicate</td> <td>Object</td> </tr> <tr> <td>this concept</td> <td>@rel value</td> <td>@qcode/@uri OR @literal OR @value</td> </tr> </table>	Subject	Predicate	Object	this concept	@rel value	@qcode/@uri OR @literal OR @value													
	Subject	Predicate	Object																	
this concept	@rel value	@qcode/@uri OR @literal OR @value																		
<p>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:</p> <table border="0"> <thead> <tr> <th>Attribute</th> <th>Attribute</th> <th>Attribute</th> <th>Applicable use case</th> </tr> <tr> <th>qcode/uri</th> <th>literal</th> <th>value</th> <th></th> </tr> </thead> <tbody> <tr> <td>1 present</td> <td>must not be present</td> <td>must not be present</td> <td>The value is a concept from a controlled vocabulary</td> </tr> <tr> <td>2 must not be present</td> <td>present</td> <td>must not be present</td> <td>The value is a concept which is not from a controlled vocabulary</td> </tr> <tr> <td>3 must not be present</td> <td>must not be present</td> <td>present</td> <td>The value is not a concept.</td> </tr> </tbody> </table> <p>In a more formal way: the use of the attributes qcode/uri, literal and value is mutually exclusive.</p>	Attribute	Attribute	Attribute	Applicable use case	qcode/uri	literal	value		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.
Attribute	Attribute	Attribute	Applicable use case																	
qcode/uri	literal	value																		
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.																	
<p>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.</p>																				
<p>Example 1 for Values: Expressing a recommendation from an analyst:</p> <pre><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></pre> <p>Example 2 for Values: Expressing some details about a sports game:</p> <pre><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:P14799" /> <related rel="crel:goalScorer" qcode="playerList:P19832" /> ... </concept></pre>																				



14.6.180 Remote Catalog Reference

Table 194. 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)	<ul style="list-style-type: none"> ▪ href (1); IRIType (page 344); A hyperlink to a remote catalog. 																	
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
<ul style="list-style-type: none"> ▪ title {itemMeta} (page 208) (0..unbounded) 																		
Child Element(s)																		
XML Schema Note(s)																		
Example(s)																		



14.6.181 Remote Content

Table 195. 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)	<p>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.</p> <p>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).</p>		
Implementation Note(s)	If the Width Unit and/or Height Unit IS NOT present, the default value(s) in Table 287 on page 367 MUST be assumed		
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values	Mandatory IPTC NewsCodes: http://cv.iptc.org/newscodes/dimensionunit/		
Attribute(s)	▪ newsContentAttributes (page 364)	Name	Datatype
		id (0..1)	XML Schema ID
		rendition (0..1) renditionuri (0..1)	QCodeType IRIType
		generator (0..1)	XML Schema string
		generated (0..1)	DateOptTimeType
		hascontent (0..1)	XML Schema boolean
	▪ targetResourceAttributes (page 365)	Name	Datatype
		href (0..1)	IRIType
		residref (0..1)	XML Schema string
		version (0..1)	XML Schema positiveInteger
		contenttype (0..1)	XML Schema string
		contenttypevariant (0..1)	XML Schema string
		format (0..1) formaturi (0..1)	QCodeType IRIType
		size (0..1)	XML Schema non NegativeInteger
	title (0..1)	XML Schema string	
	▪ timeValidityAttributes (page 361)	Name	Data Type
		validfrom (0..1) validto (0..1)	DateOptTimeType DateOptTimeType



Table 195. Remote Content (Continued)

Attribute(s) (continued)	▪ newsContentCharacteristics (page 365) (all: 0..1)	Name	Datatype
		wordcount	XML Schema nonNegativeInteger
		linecount	XML Schema nonNegativeInteger
		pagecount	XML Schema nonNegativeInteger
		width	XML Schema nonNegativeInteger
		widthunit widthunituri	QCodeType IRIType
		height	XML Schema nonNegativeInteger
		heightunit heightunituri	QCodeType IRIType
		orientation	XML S nonNegativeInteger
		layoutorientation layoutorientationuri	QCodeType IRIType
		colourspace colourspaceuri	QCodeType IRIType
		colourindicator colourindicatoruri	QCodeType IRIType
		colourdepth	XML Schema nonNegativeInteger
		resolution	XML Schema positiveInteger
		duration	XML Schema string
		durationunit durationunituri	QCodeType IRIType
		audiocodec audiocodec uri	QCodeType IRIType
		audiobitrate	XML Schema positiveInteger
		audiovbr	XML Schema boolean
		audiosamplesize	XML Schema positiveInteger
		audiosamplerate	XML Schema positiveInteger
		audiochannels audiochannelsuri	QCodeType IRIType
		videocodec videocodec uri	QCodeType IRIType
		videoavgbitrate	XML Schema positiveInteger
		videovbr	XML Schema boolean
		videoframerate	XML Schema decimal
		videoscan	enum progressive/interlaced
		videospectratio	XML Schema normalizedString
		videosampling	XML Schema normalizedString
		videoscaling videoscalinguri	QCodeType IRIType
		videodefinition videodefinitionuri	QCodeType IRIType
		▪ language (0..1); XML Schema language; The language used by the remote content.	



Table 195. Remote Content (Continued)

Child Element(s)	<ul style="list-style-type: none"> ▪ altLoc (page 74) (0..unbounded) ▪ altId (page 73) (0..unbounded) ▪ channel {News Item} (page 89) (0..unbounded) ▪ signal (page 292) (0..unbounded) ▪ remoteContentExtProperty (page 274) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	<pre> <!-- RC: Picture, using implicit default dimensionunit:pixels --> <remoteContent residref="tag:reuters.com,0000:binary_BTRE4A31LE800-THUMBNAI" 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-THUMBNAI" rendition="rend:thumbnail" contenttype="image/gif" format="fmt:gif87a" width="100" widthunit="dimensionunit:points" height="100" heightunit="dimensionunit:points"/> </pre>



14.6.182 Remote Content Extension Property

Table 196. Remote Content Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	remoteContentExtProperty
Title	Remote Content Extension Property
Definition	A generic extension property for remote content; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.183 Remote Information about a Concept

Table 197. Remote Information about a Concept

(XML) Data Model	Element
Namespace (prefix)	nar
Name	remotelInfo
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 346)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.184 Rights Information

Table 198. 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 or parts of the Item.							
User Note(s)								
Implementation Note(s)								
XML Schema Spec	At: PCL							
Datatype								
Internally Ctrl Values								
Externally Ctrl Values								
Attribute(s)	<ul style="list-style-type: none"> ▪ 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 244) element which in turn references the part of the content. 							
	<ul style="list-style-type: none"> ▪ scope; (0..1), QCodeListType; scopeuri (0..1); IRIListType; 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/. 							
	<ul style="list-style-type: none"> ▪ aspect; (0..1), QCodeListType; Indicates by a QCode 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/riaspect/. ▪ aspecturi; (0..1), IRIType; Indicates by a URI 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/riaspect/. ▪ 							
	<ul style="list-style-type: none"> ▪ timeValidityAttributes (page 361) 		<table border="1"> <thead> <tr> <th>Name</th> <th>Data Type</th> </tr> </thead> <tbody> <tr> <td>validfrom (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>validto (0..1)</td> <td>DateOptTimeType</td> </tr> </tbody> </table>	Name	Data Type	validfrom (0..1)	DateOptTimeType	validto (0..1)
Name	Data Type							
validfrom (0..1)	DateOptTimeType							
validto (0..1)	DateOptTimeType							
Child Element(s)	<ul style="list-style-type: none"> ▪ accountable (page 69) (0..1) 							
	<ul style="list-style-type: none"> ▪ copyrightHolder (page 121) (0..1) 							
	<ul style="list-style-type: none"> ▪ copyrightNotice (page 122) (0..unbounded) 							
	<ul style="list-style-type: none"> ▪ usageTerms (page 304) (0..unbounded) 							
	<ul style="list-style-type: none"> ▪ link (page 214) (0..unbounded) 							
	<ul style="list-style-type: none"> ▪ rightsInfoExtProperty (page 277) (0..unbounded) 							
	<ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 							
XML Schema Note(s)								
Example(s)								



14.6.185 Rights Information Extension Property

Table 199. Rights Information Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rightsInfoExtProperty
Title	Rights Information Extension Property
Definition	A generic extension property for rights information; the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.186 Role in the Content Stream

Table 200. 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	Qualified Property Type (page 352)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.187 Role in the Workflow

Table 201. 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	Qualified Property Type (page 352)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.188 Ruby

Table 202. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
	<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	Name	Datatype
		xml:lang (0..1)	XML Schema language
dir (0..1)		XML Schema string: enumeration <i>ltr, rtl</i> .	
Child Element(s)	<ul style="list-style-type: none"> ▪ rb (page 281) (1) 		
	<ul style="list-style-type: none"> ▪ rp (page 282) (see XML Schema note below) 		
	<ul style="list-style-type: none"> ▪ rt (page 283) (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 <i>rt</i> element or a single <i>rp-rt-rp</i> sequence of elements. Ruby parentheses (<rp>, empty elements) must be used as a pair: either both are present or none is present.		
Example(s)	<pre><ruby> <rb>IPTC</rb> <rp>(<rp/><rt>International Press Telecommunications Council</rt><rp></rp> </ruby></pre>		



14.6.189 Ruby Base

Table 203. 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 280).
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)	



14.6.190 Ruby Parenthesis

Table 204. 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 280).
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)	<pre><ruby> <rb>IPTC</rb> <rp>(<rp/><rt>International Press Telecommunications Council</rt><rp></rp> </ruby></pre>



14.6.191 Ruby Text

Table 205. Ruby Text

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rt
Title	Ruby Text
Definition	Ruby text.
User Note(s)	Also see ruby (page 280).
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)	



14.6.192 Same As

Table 206. 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 336)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> timeValidityAttributes (page 361) 	Name	Data Type
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.193 SameAs for a Scheme (DEPRECATED)

Table 207. Same As of a Scheme

(XML) Data Model	Element
Namespace (prefix)	nar
Name	sameAs {Scheme} (DEPRECATED)
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)	This element SHOULD NOT be used in NewsML-G2 2.11 and higher , the element sameAsScheme should be used instead.
XML Schema Spec	At: Both CCL and PCL
Datatype	IRIType (page 344)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.194 SameAs Scheme

Table 208. Same As of a Scheme

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	sameAsScheme		
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)	This element replaces the sameAs element as child of a scheme element.		
XML Schema Spec	At: PCL		
Datatype	IRIType (page 344)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

**14.6.195 Scheduled**

Table 209. 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 311)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.196 Scheme Declaration

Table 210. Scheme Declaration

(XML) Data Model	Element																	
Namespace (prefix)	nar																	
Name	scheme																	
Title	Scheme Declaration																	
Definition	A scheme alias-to-URI mapping with an optional description of the scheme.																	
User Note(s)																		
Implementation Note(s)																		
XML Schema Spec	At: PCL																	
Datatype																		
Internally Ctrl Values																		
Externally Ctrl Values																		
Attribute(s)	<ul style="list-style-type: none"> alias (1); XML Schema NCName; A short string assigned by the provider as a representation of the scheme URI. uri (1); IRIType (page 344); The URI which identifies the scheme. 																	
	<ul style="list-style-type: none"> commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
<ul style="list-style-type: none"> sameAs {Scheme} (DEPRECATED) (page 285) (0..unbounded) sameAsScheme (page 286) (0..unbounded) name (page 260) (0..unbounded) definition (page 96) (0..unbounded) note (page 101) (0..unbounded) 																		
Child Element(s)																		
XML Schema Note(s)																		
Example(s)																		
Processing Model for the sameAs child	<ul style="list-style-type: none"> * 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. 																	



14.6.197 Scheme Metadata

Table 211. Scheme Metadata

(XML) Data Model	Element																		
Namespace (prefix)	nar																		
Name	schemeMeta																		
Title	Scheme Metadata																		
Definition	Metadata about a scheme conveyed by a Knowledge Item																		
User Note(s)	Only if the itemClass of the Knowledge Item is set to cinat:scheme this element may be used.																		
Implementation Note(s)																			
XML Schema Spec	At: PCL																		
Datatype																			
Internally Ctrl Values																			
Externally Ctrl Values																			
Attribute(s)	<ul style="list-style-type: none"> ▪ .uri (1); IRIType (page 344); The URI which identifies the scheme. ▪ authority (0..1); IRIType (page 344); Defines the authority controlling the scheme. ▪ preferredalias (0..1); XML Schema string; The alias preferred by the scheme authority. 																		
	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>▪ commonPowerAttributes (page 361)</td> <td></td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	▪ commonPowerAttributes (page 361)		custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
	Name	Datatype																	
	id (0..1)	XML Schema ID																	
	creator (0..1) creatoruri (0..1)	QCodeType IRIType																	
	modified (0..1)	DateOptTimeType																	
	▪ commonPowerAttributes (page 361)																		
	custom (0..1)	XML Schema boolean																	
pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType																		
how (0..1) howuri (0..1)	QCodeType IRIType																		
why (0..1) whyuri (0..1)	QCodeType IRIType																		
Child Element(s)	<ul style="list-style-type: none"> ▪ sameAsScheme (page 286) (0..unbounded) ▪ name (page 260) (0..unbounded) ▪ definition (page 96) (0..unbounded) ▪ note (page 101) (0..unbounded) ▪ related (page 267) (0..unbounded) ▪ schemeMetaExtProperty (page 290) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 																		
XML Schema Note(s)																			
Example(s)																			



14.6.198 Scheme Metadata Extension Property

Table 212. Scheme Metadata Extension Property

(XML) Data Model	Element
Namespace (prefix)	nar
Name	schemeMetaExtProperty
Title	Scheme Metadata Extension Property
Definition	A generic extension property for scheme metadata the semantics are defined by the concept referenced by the rel attribute. The semantics of the Extension Property must have the same scope as its parent property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 2 Extension Property Type (page 328)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.199 Sender

Table 213. 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. Rule for @qcode and @uri in an element: <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 		
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 363)	Name	Datatype
		qcode (0..1)	QCodeType
		uri (0..1)	XML Schema anyURI
		literal (0..1)	NormalizedStringType (of G2)
		type typeuri	QCodeType IRIType
		role roleuri	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.200 Signal

Table 214. 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 352)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ severity (0..1); QCodeType (page 351); severityuri (0..1); IRI Type (page 344); 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)	



14.6.201 Slugline

Table 215. 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)	<i>separator</i> 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 342)					
Internally Ctrl Values						
Externally Ctrl Values						
Attribute(s)	<ul style="list-style-type: none"> separator (0..1); XML Schema string; The character string acting as a separator between the tokens in the slugline. 					
	<ul style="list-style-type: none"> role (0..1); QCodeType (page 351); roleuri (0..1); IRIType (page 344); The role this slugline plays in the scope of the full content. 					
	<ul style="list-style-type: none"> rankingAttributes (page 362) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>rank (0..1)</td> <td>XML Schema nonNegativeInteger</td> </tr> </tbody> </table>	Name	Datatype	rank (0..1)	XML Schema nonNegativeInteger
Name	Datatype					
rank (0..1)	XML Schema nonNegativeInteger					
Child Element(s)						
XML Schema Note(s)						
Example(s)						



14.6.202 Span

Table 216. 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)	<ul style="list-style-type: none"> ▪ class (0..1); XML Schema NMTOKENS; List of classes. 																	
	▪ i18nAttributes (page 360)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr</i>, <i>rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .										
		Name	Datatype															
		xml:lang (0..1)	XML Schema language															
	dir (0..1)	XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .																
	▪ commonPowerAttributes (page 361)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
custom (0..1)		XML Schema boolean																
pubconstraint (0..1) pubconstrainturi (0..1)		QCodeListType IRIListType																
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
<ul style="list-style-type: none"> ▪ Ruby (page 280) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 																		
Child Element(s)																		
XML Schema Note(s)																		
Example(s)																		



14.6.203 Start Date/Time

Table 217. 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 311)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.204 Subject

Table 218. 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 323)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ rankingAttributes (page 362)	Name	Datatype
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.205 Time Delimiter

Table 219. 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)	<p>The time unit may take the following values, taken from an IPTC defined controlled vocabulary:</p> <ul style="list-style-type: none"> - <i>timecode</i>: An SMPTE timecode containing a string encoded identification. Timestamp format: hh:mm:ss:ff (ff for frames). - <i>timeCodeDropFrame</i>: An SMPTE timecode containing a string encoded identification. Timestamp format: hh:mm:ss:ff (ff for frames). The drop frame flag should be set. - <i>editUnit</i>: 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. - <i>normalPlayTime</i>: Indicates the position relative to the beginning of the presentation. Timestamp format: hh:mm:ss.mmm (mmm for milliseconds). See also: RFC 2326. - <i>seconds</i>: Time given in full seconds. Timestamp format: long unsigned integer. - <i>milliseconds</i>: Time given in full milliseconds. Timestamp format: long unsigned integer.
Implementation Note(s)	<p>If a time unit IS NOT present, the value <i>editUnit</i> 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'.</p>
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	Mandatory IPTC NewsCodes: http://cv.iptc.org/newscodes/timeunit/



Table 219. Time Delimiter

Attribute(s)	<ul style="list-style-type: none"> ▪ 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 (0..1); QCodeType (page 351); timeunituri (0..1); IRIType (page 344); The unit used for the start and end timestamps. The use of either the timeunit or the timeunituri attribute is mandatory (from NewsML-G2 v 2.19 onwards) ▪ renditionref (0..1); QCodeType (page 351); Refers to the content rendition with this QCode as rendition attribute value - expressed by a QCode ▪ renditionrefiuri (0..1); IRI Type (page 344); Refers to the content rendition with this QCode as rendition attribute value - expressed by a URI 																	
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
Child Element(s)																		
XML Schema Note(s)																		
Example(s)	<pre> <!-- 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" timeunit="timeunit:normalPlayTime"/> </partMeta> </pre>																	



14.6.206 Timestamp

Table 220. 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 320)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> 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)	



14.6.207 Transmission Identifier

Table 221. 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)	



14.6.208 Type of a Concept

Table 222. 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 352)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.209 User Interaction

Table 223. User Interaction

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	userInteraction		
Title	User Interaction		
Definition	Reflects a specific kind of user interaction with the content of this item.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
	why (0..1) whyuri (0..1)	QCodeType IRIType	
	<ul style="list-style-type: none"> ▪ interactions (1); XML Schema NonNegativeInteger; The count of executed interactions. 		
	<ul style="list-style-type: none"> ▪ interactiontype (0..1); QCodeType; interactiontypeuri (0..1); IRIType; The type of interaction which is reflected by this property. The use of either the interactiontype or the interactiontypeuri attribute is mandatory (from NewsML-G2 v 2.19 onwards) (Examples: Facebook's Likes, Google's +1, Twitter Retweets, ...) 		
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.210 Urgency

Table 224. 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 341)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.6.211 Usage Terms

Table 225. 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 355)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.6.212 Visual Region Delimiter

Table 226. 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)	<ul style="list-style-type: none"> ▪ 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. 																	
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
Child Element(s)																		
XML Schema Note(s)																		
Example(s)																		



14.6.213 Web Address

Table 227. 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 344)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the web address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.7 Element Group Definitions

14.7.1 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 228. Concept Definition Group Elements

Element Title	Element Name	Card	Described on Page
Concept Name	name	(0..unbounded)	100
Concept Definition	definition	(0..unbounded)	96
Note	note	(0..unbounded)	101
Facet	facet	(0..unbounded)	165
Remote Information	remotelInfo	(0..unbounded)	275
Hierarchy Info	hierarchyInfo	(0..unbounded)	

14.7.2 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 229. Concept Relationships Group Elements

Element Title	Element Name	Card	Described on Page
Same As	sameAs {Relationship}	(0..unbounded)	284
Broader	broader	(0..unbounded)	83
Narrower	narrower	(0..unbounded)	222
Related	related	(0..unbounded)	267

14.7.3 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 230. Entity Details Group Elements

Element Title	Element Name	Card	Described on Page
Person Details	personDetails	(1)	248
Organisation Details	organisationDetails	(1)	238
Geopolitical Area Details	geoAreaDetails	(1)	173
POI Details	POIDetails	(1)	253
Object Details	objectDetails	(1)	234



14.7.4 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 231. Administrative Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Urgency	urgency	(0..1)	303
Date Content Created	contentCreated	(0..1)	129
Date Content Modified	contentModified	(0..1)	130
Located	located	(0..unbounded)	200
Information Source	infoSource	(0..unbounded)	303
Creator	creator	(0..unbounded)	125
Contributor	contributor	(0..unbounded)	120
Audience	audience	(0..unbounded)	80
Rating	rating	(0..unbounded)	262
User Interaction	userInteraction	(0..unbounded)	302
Excluded Audience	exclAudience	(0..unbounded)	161
Alternative Identifier	altId	(0..unbounded)	73

14.7.5 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 232. Knowledge Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Language	language	(0..unbounded)	211
Keyword	keyword	(0..unbounded)	209
Subject	subject	(0..unbounded)	296
Description	description	(0..unbounded)	146

14.7.6 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 233. Descriptive Metadata Core Group Elements

Element Title	Element Name	Card	Described on Page
Language	language	(0..unbounded)	211
Keyword	keyword	(0..unbounded)	209
Subject	subject	(0..unbounded)	296
Slugline	slugline	(0..unbounded)	293
Headline	headline	(0..unbounded)	182
Description	description	(0..unbounded)	146



14.7.7 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 234. Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Language	language	(0..unbounded)	211
Genre	genre	(0..unbounded)	171
Keyword	keyword	(0..unbounded)	209
Subject	subject	(0..unbounded)	296
Slugline	slugline	(0..unbounded)	293
Headline	headline	(0..unbounded)	182
Dateline	dateline	(0..unbounded)	143
By	by	(0..unbounded)	84
CreditLine	creditline	(0..unbounded)	126
Description	description	(0..unbounded)	146

14.7.8 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 235. Item Management Group Elements

Element Title	Element Name	Card	Described on Page
Item Class	itemClass	(1)	201
Content Provider	provider	(1)	118
Date Item Version Created	versionCreated	(1)	133
Date Item First Created	firstCreated	(0..1)	132
Date Item Embargo Ends	embargoed	(0..1)	131
Publish Status	pubStatus	(0..1)	261
Role in the Workflow	role	(0..1)	279
File Name	filename	(0..1)	167
Generator Tool	generator	(0..unbounded)	170
Profile	profile	(0..1)	259
Editorial Service	service	(0..unbounded)	154
Item Title	title {itemMeta}	(0..unbounded)	208
Editorial Note	edNote	(0..unbounded)	153
Member Of	memberOf	(0..unbounded)	219
Instance Of	instanceOf	(0..unbounded)	198
Signal	signal	(0..unbounded)	292
Alternative Representation	altRep	(0..unbounded)	75
Deliverable Of	deliverableOf	(0..unbounded)	147
Hash Value	hash	(0..unbounded)	147
Expiry Date	expires	(0..unbounded)	164



14.8 Datatype Definitions

14.8.1 Any Item Type

Table 236. 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)	<ul style="list-style-type: none"> ▪ standard; (0..1); string value: default = "XML Schema string"; The IPTC standard to which the Item is conformant. 						
	<ul style="list-style-type: none"> ▪ standardversion; (1); XML Schema string; restricted to the format "integer.integer"; The major-minor version of the XML schema specifying the Item. 						
	<ul style="list-style-type: none"> ▪ conformance; (1); string value: fixed = "XML Schema string" - default = "core"; The conformance level to which the Item is conformant. 						
	<ul style="list-style-type: none"> ▪ guid; (1); XML Schema string; The persistent, universally unique identifier for the Item. 						
	<ul style="list-style-type: none"> ▪ version; (0..1); XML Schema positiveInteger; The version of the Item. 						
	<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)
Name	Datatype						
xml:lang (0..1)	XML Schema language						
dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .						
Child Element(s)	<ul style="list-style-type: none"> ▪ W3C XML Digital Signature (from namespace http://www.w3.org/2000/09/xmldsig#) (0..1) 						
	<ul style="list-style-type: none"> ▪ catalogRef (page 270) (0..unbounded) 						
	<ul style="list-style-type: none"> ▪ catalog (page 86) (0..unbounded) 						
	<ul style="list-style-type: none"> ▪ hopHistory (page 185) (0..1) 						
	<ul style="list-style-type: none"> ▪ rightsInfo (page 276) (0..unbounded) ▪ itemMeta (page 203) (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)							

14.8.2 Approximate Date and Time Property Type

Table 237. 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)	<ul style="list-style-type: none"> ▪ approxstart (0..1); TruncatedDateTimeType (page 357); The date (and optionally time) at which the approximation range begins. ▪ approxend (0..1); TruncatedDateTimeType (page 357); The date (and optionally time) at which the approximation range ends. 																	
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
Child Element(s)																		
XML Schema Note(s)																		
Example(s)	<p>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.</p>																	



14.8.3 Audience Type

Table 238. Audience Type

(XML) Data Model	Type											
Namespace (prefix)	nar											
Name	AudienceType											
Title	Audience Type											
Definition	An audience for the content.											
User Note(s)	<i>significance: 1</i> – corresponds to the highest significance. <i>significance: 9</i> – corresponds to the lowest significance.											
Implementation Note(s)												
XML Schema Spec	At: PCL											
Datatype	Extends Flex1PropType (page 326)											
Internally Ctrl Values												
Externally Ctrl Values												
Attribute(s)	<ul style="list-style-type: none"> ▪ <i>significance</i> (0..1); Int1To9Type (page 341). A qualifier which indicates the expected significance of the content for this specific audience. 											
	▪ quantifyAttributes (page 363)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>confidence (0..1)</td> <td>Int100Type</td> </tr> <tr> <td>relevance (0..1)</td> <td>Int100Type</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>derivedfrom (0..1) DEPRECATED</td> <td>QCodeType</td> </tr> </tbody> </table>	Name	Datatype	confidence (0..1)	Int100Type	relevance (0..1)	Int100Type	why (0..1) whyuri (0..1)	QCodeType IRIType	derivedfrom (0..1) DEPRECATED	QCodeType
		Name	Datatype									
		confidence (0..1)	Int100Type									
		relevance (0..1)	Int100Type									
why (0..1) whyuri (0..1)	QCodeType IRIType											
derivedfrom (0..1) DEPRECATED	QCodeType											
Child Element(s)												
XML Schema Note(s)												
Example(s)												



14.8.4 Block Type

Table 239. 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)	<ul style="list-style-type: none"> role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); 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. 		
	<ul style="list-style-type: none"> commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
	why (0..1) whyuri (0..1)	QCodeType IRIType	
	<ul style="list-style-type: none"> i18nAttributes (page 360) 	Name	Datatype
xml:lang (0..1)		XML Schema language	
	dir (0..1)	XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .	
Child Element(s)	<ul style="list-style-type: none"> a (page 76) (0..unbounded) span (page 294) (0..unbounded) ruby (page 280) (0..unbounded) br (page 213) (0..unbounded) inline (page 189) (0..unbounded) Extension Point (0..unbounded). Any set of provider-defined properties. 		



Table 239. Block Type (Continued)

XML Schema Note(s)	
Example(s)	



14.8.5 Concept Identifier Type

Table 240. 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)	Rule for @qcode and @uri in an element: If both attributes, @qcode and @uri, are present the @qcode takes precedence.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code which identifies a concept. ▪ uri (0..1), XML Schema anyURI; A URI which identifies a concept. The use of either the qcode or the uri attribute is mandatory (from NewsML-G2 v 2.19 onwards) ▪ created (0..1); DateOptTimeType (page 318); The date (and optionally the time) when the identifier was created. ▪ retired (0..1); DateOptTimeType (page 318); 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)	



14.8.6 Content Metadata Type

Table 241. 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 360)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
Child Element(s)	▪ Administrative Metadata Group (page 308) (0..1)	Element Name	Page
		urgency (0..1)	303
		contentCreated (0..1)	129
		contentModified (0..1)	130
		located (0..unbounded)	216
		infoSource (0..unbounded)	200
		creator (0..unbounded)	125
		contributor (0..unbounded)	120
		audience (0..unbounded)	80
		exclAudience (0..unbounded)	161
	altId (0..unbounded)	73	
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			



14.8.7 Date and Optional Time Property Type

Table 242. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.8.8 Date and Optional Time Type

Table 243. 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)	

14.8.9 Date and Time or Null Value Property Type

Table 244. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

14.8.10 Date and Time Property Type

Table 245. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.8.11 Electronic Address Type

Table 246. *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)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the electronic address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.8.12 Electronic Address Tech Type

Table 247. 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)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the electronic address. ▪ tech (0..1); QCodeType (page 351); techuri (0..1); IRI Type (page 344); The technical variant of the electronic address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.8.13 Flexible 1 Concept Property Type

Table 248. 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 326)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ quantifyAttributes (page 363)	Name	Datatype
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
		why (0..1) whyuri (0..1)	QCodeType IRIType
		derivedfrom (0..1) DEPRECATED	QCodeType
Child Element(s)	▪ bag (page 81) (0..1)		
XML Schema Note(s)			
Example(s)			



14.8.14 Flexible 1 Extension Property Type

Table 249. Flexible 1 Extension Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Flex1ExtPropType
Title	Flexible 1 Extension Property Type
Definition	Flexible generic PCL-type for controlled, uncontrolled values and arbitrary values recommended as datatype for proprietary properties of Extension Points
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 1 Property Type (page 326)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ value; (0..1); XML Schema String; The related property's value (see also: qcode, uri 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 351); valueunituri; (0.1); IRI Type (page 344); The unit of the value attribute.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

14.8.15 Flexible 1 Party Property Type

Table 250. 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 332)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); A refinement of the semantics of the property.
Child Element(s)	



Table 250. Flexible 1 Party Property Type

XML Schema Note(s)	
Example(s)	



14.8.16 Flexible 1 Property Type

Table 251. 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)	<p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 																	
Implementation Note(s)																		
XML Schema Spec	At: PCL																	
Datatype																		
Internally Ctrl Values																		
Externally Ctrl Values																		
	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. (For an inlineRef element this is the identifier of the described concept.) <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. (For an inlineRef element this is the identifier of the described concept.) <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. (For an inlineRef element this is the identifier of the described concept.) <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p>																	
	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as controlled or uncontrolled property value. 																	
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
id (0..1)		XML Schema ID																
creator (0..1) creatoruri (0..1)		QCodeType IRIType																
modified (0..1)		DateOptTimeType																
custom (0..1)		XML Schema boolean																
pubconstraint (0..1) pubconstrainturi (0..1)		QCodeListType IRIListType																
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .											
	Name	Datatype																
	xml:lang (0..1)	XML Schema language																
dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .																	



Table 251. Flexible 1 Property Type (Continued)

Child Element(s)	▪ Concept Definition Group (page 307) (0..1)	Element Name	Page
		name (0..unbounded)	100
		definition (0..unbounded)	96
		facet (0..unbounded)	165
		remoteInfo (0..unbounded)	275
		note (0..unbounded)	101
	▪ Concept Relationships Group (page 307) (0..1)	Element Name	Page
		broader (0..unbounded)	83
		narrower (0..unbounded)	222
		related (0..unbounded)	267
		sameAs {Relationship} (0..unbounded)	284
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
	XML Schema Note(s)		
	Example(s)		



14.8.17 Flexible 2 Extension Property Type

Table 252. Flexible 2 Extension Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Flex2ExtPropType
Title	Flexible 2 Extension Property Type
Definition	Flexible generic PCL-Type for controlled, uncontrolled values and arbitrary values, with a mandatory relationship.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Flexible 1 Extension Property Type (page 324)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ rel (0..1); QCode Type (page 351); defining the semantics of this property - expressed by a QCode ▪ reluri (0..1); IRI Type (page 344); defining the semantics of this property - expressed by a URI <p>The use of either the rel or the reluri attribute is mandatory (from NewsML-G2 v 2.19 onwards)</p> <ul style="list-style-type: none"> ▪ validfrom (0..1); Date and Optional Time Type (page 318); The date (and, optionally, the time) before which a relationship is not valid. ▪ validto (0..1); Date and Optional Time Type (page 318); The date (and, optionally, the time) after which a relationship is not valid.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

14.8.18 Flexible Location Property Type

Table 253. 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)	<p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	



Table 253. Flexible Location Property Type (Continued)

	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p>																									
	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); <p>The type of the concept assigned as controlled or uncontrolled property value.</p>																									
Attribute(s)	<table border="1"> <thead> <tr> <th></th> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td rowspan="7">▪ commonPowerAttributes (page 361)</td> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td rowspan="3">▪ i18nAttributes (page 360)</td> <td>Name</td> <td>Datatype</td> </tr> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>		Name	Datatype	▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType	▪ i18nAttributes (page 360)	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
		Name	Datatype																							
▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID																								
	creator (0..1) creatoruri (0..1)	QCodeType IRIType																								
	modified (0..1)	DateOptTimeType																								
	custom (0..1)	XML Schema boolean																								
	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType																								
	how (0..1) howuri (0..1)	QCodeType IRIType																								
	why (0..1) whyuri (0..1)	QCodeType IRIType																								
▪ i18nAttributes (page 360)	Name	Datatype																								
	xml:lang (0..1)	XML Schema language																								
	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .																								
Child Element(s)	<table border="1"> <thead> <tr> <th></th> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td rowspan="5">▪ Concept Definition Group (page 307) (0..1)</td> <td>name (0..unbounded)</td> <td>100</td> </tr> <tr> <td>definition (0..unbounded)</td> <td>96</td> </tr> <tr> <td>facet (0..unbounded)</td> <td>165</td> </tr> <tr> <td>remoteInfo (0..unbounded)</td> <td>275</td> </tr> <tr> <td>note (0..unbounded)</td> <td>101</td> </tr> <tr> <td rowspan="4">▪ Concept Relationships Group (page 307) (0..1)</td> <td>broader (0..unbounded)</td> <td>83</td> </tr> <tr> <td>narrower (0..unbounded)</td> <td>222</td> </tr> <tr> <td>related (0..unbounded)</td> <td>267</td> </tr> <tr> <td>sameAs {Relationship} (0..unbounded)</td> <td>284</td> </tr> </tbody> </table> <ul style="list-style-type: none"> ▪ geoAreaDetails (page 173) (0..1) <p>Or</p> <ul style="list-style-type: none"> ▪ POIDetails (page 253) (0..1) <ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 		Element Name	Page	▪ Concept Definition Group (page 307) (0..1)	name (0..unbounded)	100	definition (0..unbounded)	96	facet (0..unbounded)	165	remoteInfo (0..unbounded)	275	note (0..unbounded)	101	▪ Concept Relationships Group (page 307) (0..1)	broader (0..unbounded)	83	narrower (0..unbounded)	222	related (0..unbounded)	267	sameAs {Relationship} (0..unbounded)	284		
	Element Name	Page																								
▪ Concept Definition Group (page 307) (0..1)	name (0..unbounded)	100																								
	definition (0..unbounded)	96																								
	facet (0..unbounded)	165																								
	remoteInfo (0..unbounded)	275																								
	note (0..unbounded)	101																								
▪ Concept Relationships Group (page 307) (0..1)	broader (0..unbounded)	83																								
	narrower (0..unbounded)	222																								
	related (0..unbounded)	267																								
	sameAs {Relationship} (0..unbounded)	284																								
XML Schema Note(s)																										
Example(s)																										



14.8.19 Flexible Organisation Property Type

Table 254. 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)	<p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 																
Implementation Note(s)																	
XML Schema Spec	At: PCL																
Datatype																	
Internally Ctrl Values																	
Externally Ctrl Values																	
	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p> <ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as controlled or uncontrolled property value. 																
Attribute(s)	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>▪ commonPowerAttributes (page 361)</td> <td>custom (0..1) XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	▪ commonPowerAttributes (page 361)	custom (0..1) XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
	Name	Datatype															
id (0..1)	XML Schema ID																
creator (0..1) creatoruri (0..1)	QCodeType IRIType																
modified (0..1)	DateOptTimeType																
▪ commonPowerAttributes (page 361)	custom (0..1) XML Schema boolean																
pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType																
how (0..1) howuri (0..1)	QCodeType IRIType																
why (0..1) whyuri (0..1)	QCodeType IRIType																
<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>▪ i18nAttributes (page 360)</td> <td>xml:lang (0..1) XML Schema language</td> </tr> <tr> <td></td> <td>dir (0..1) XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	▪ i18nAttributes (page 360)	xml:lang (0..1) XML Schema language		dir (0..1) XML Schema string: enumeration <i>ltr, rtl</i> .											
Name	Datatype																
▪ i18nAttributes (page 360)	xml:lang (0..1) XML Schema language																
	dir (0..1) XML Schema string: enumeration <i>ltr, rtl</i> .																



Table 254. Flexible Organisation Property Type

	<ul style="list-style-type: none"> ▪ organisationDetails (page 238) (0..1) 													
Child Element(s)	<ul style="list-style-type: none"> ▪ Concept Definition Group (page 307) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>name (0..unbounded)</td> <td>100</td> </tr> <tr> <td>definition (0..unbounded)</td> <td>96</td> </tr> <tr> <td>facet (0..unbounded)</td> <td>165</td> </tr> <tr> <td>remoteInfo (0..unbounded)</td> <td>275</td> </tr> <tr> <td>note (0..unbounded)</td> <td>101</td> </tr> </tbody> </table>	Element Name	Page	name (0..unbounded)	100	definition (0..unbounded)	96	facet (0..unbounded)	165	remoteInfo (0..unbounded)	275	note (0..unbounded)	101
		Element Name	Page											
		name (0..unbounded)	100											
		definition (0..unbounded)	96											
		facet (0..unbounded)	165											
	remoteInfo (0..unbounded)	275												
	note (0..unbounded)	101												
	<ul style="list-style-type: none"> ▪ Concept Relationships Group (page 307) (0..1) 	<table border="1"> <thead> <tr> <th>Element Name</th> <th>Page</th> </tr> </thead> <tbody> <tr> <td>broader (0..unbounded)</td> <td>83</td> </tr> <tr> <td>narrower (0..unbounded)</td> <td>222</td> </tr> <tr> <td>related (0..unbounded)</td> <td>267</td> </tr> <tr> <td>sameAs {Relationship} (0..unbounded)</td> <td>284</td> </tr> </tbody> </table>	Element Name	Page	broader (0..unbounded)	83	narrower (0..unbounded)	222	related (0..unbounded)	267	sameAs {Relationship} (0..unbounded)	284		
		Element Name	Page											
		broader (0..unbounded)	83											
narrower (0..unbounded)		222												
related (0..unbounded)	267													
sameAs {Relationship} (0..unbounded)	284													
<ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 														
XML Schema Note(s)														
Example(s)														

14.8.20 Flexible Party Property Type

Table 255. 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)	<p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 																	
Implementation Note(s)																		
XML Schema Spec	At: PCL																	
Datatype																		
Internally Ctrl Values																		
Externally Ctrl Values																		
	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p>																	
	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); <p>The type of the concept assigned as controlled or uncontrolled property value.</p>																	
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
Name	Datatype																	
id (0..1)	XML Schema ID																	
creator (0..1) creatoruri (0..1)	QCodeType IRIType																	
modified (0..1)	DateOptTimeType																	
custom (0..1)	XML Schema boolean																	
pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType																	
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
	<ul style="list-style-type: none"> ▪ i18nAttributes (page 360) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .										
Name	Datatype																	
xml:lang (0..1)	XML Schema language																	
dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .																	



Table 255. Flexible Party Property Type (Continued)

Child Element(s)	<ul style="list-style-type: none"> ▪ personDetails (page 248) (0..1) or ▪ organisationDetails (page 238) (0..1) 		
	<ul style="list-style-type: none"> ▪ Concept Definition Group (page 307) (0..1) 	Element Name	Page
		name (0..unbounded)	100
		definition (0..unbounded)	96
		facet (0..unbounded)	165
		remoteInfo (0..unbounded)	275
	<ul style="list-style-type: none"> ▪ Concept Relationships Group (page 307) (0..1) 	Element Name	Page
		broader (0..unbounded)	83
		narrower (0..unbounded)	222
		related (0..unbounded)	267
	sameAs {Relationship} (0..unbounded)	284	
	<ul style="list-style-type: none"> ▪ Extension Point (0..unbounded). Any set of provider-defined properties. 		
XML Schema Note(s)			
Example(s)			



14.8.21 Flex Person Property Type

Table 256. 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)	Rule for @qcode and @uri in an element: - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence.																	
Implementation Note(s)																		
XML Schema Spec	At: PCL																	
Datatype																		
Internally Ctrl Values																		
Externally Ctrl Values																		
	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. Or <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. Or <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))																	
	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as controlled or uncontrolled property value. 																	
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1)</td> <td>QCodeType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1)	QCodeType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
id (0..1)		XML Schema ID																
creator (0..1)		QCodeType																
modified (0..1)		DateOptTimeType																
custom (0..1)		XML Schema boolean																
pubconstraint (0..1) pubconstrainturi (0..1)		QCodeListType IRIListType																
how (0..1) howuri (0..1)		QCodeType IRIType																
why (0..1) whyuri (0..1)	QCodeType IRIType																	
	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .											
Name	Datatype																	
xml:lang (0..1)	XML Schema language																	
dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .																	



Table 256. Flex Person Property Type (Continued)

Child Element(s)	▪ personDetails (page 248) (0..1)		
	▪ Concept Definition Group (page 307) (0..1)	Element Name	Page
		name (0..unbounded)	100
		definition (0..unbounded)	96
		facet (0..unbounded)	165
		remoteInfo (0..unbounded)	275
		note (0..unbounded)	101
	▪ Concept Relationships Group (page 307) (0..1)	Element Name	Page
		broader (0..unbounded)	83
		narrower (0..unbounded)	222
related (0..unbounded)		267	
	sameAs {Relationship} (0..unbounded)	284	
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			



14.8.22 Flexible Property Type

Table 257. 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)	<p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 																												
Implementation Note(s)																													
XML Schema Spec	At: PCL																												
Datatype																													
Internally Ctrl Values																													
Externally Ctrl Values																													
	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p> <ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as a controlled or uncontrolled property value. 																												
Attribute(s)	<table border="1"> <thead> <tr> <th></th> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td rowspan="8">▪ commonPowerAttributes (page 361)</td> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td></td> <td></td> <td></td> </tr> <tr> <td rowspan="3">▪ i18nAttributes (page 360)</td> <th>Name</th> <th>Datatype</th> </tr> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>		Name	Datatype	▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType				▪ i18nAttributes (page 360)	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
	Name	Datatype																											
▪ commonPowerAttributes (page 361)	id (0..1)	XML Schema ID																											
	creator (0..1) creatoruri (0..1)	QCodeType IRIType																											
	modified (0..1)	DateOptTimeType																											
	custom (0..1)	XML Schema boolean																											
	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType																											
	how (0..1) howuri (0..1)	QCodeType IRIType																											
	why (0..1) whyuri (0..1)	QCodeType IRIType																											
▪ i18nAttributes (page 360)	Name	Datatype																											
	xml:lang (0..1)	XML Schema language																											
	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .																											
Child Element(s)	<ul style="list-style-type: none"> ▪ name (page 260) (0..unbounded) ▪ hierarchyInfo (page 183) (0..unbounded) 																												



Table 257. Flexible Property Type

XML Schema Note(s)	
Example(s)	



14.8.23 Flexible Property 2 Type

Table 258. Flexible Property 2 Type

(XML) Data Model	Type																	
Namespace (prefix)	nar																	
Name	FlexProp2Type																	
Title	Flexible Property 2 Type																	
Definition	Flexible generic data type for both controlled and uncontrolled values, variant 2																	
User Note(s)	<p>Rule for @qcode and @uri in an element:</p> <ul style="list-style-type: none"> - An element SHOULD NOT use both a @qcode and a @uri. - If both attributes, @qcode and @uri, are present the @qcode takes precedence. 																	
Implementation Note(s)																		
XML Schema Spec	At: PCL																	
Datatype																		
Internally Ctrl Values																		
Externally Ctrl Values																		
	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ uri, (0..1); XML Schema anyURI; A URI assigned as identifier of the property value. <p>Or</p> <ul style="list-style-type: none"> ▪ literal (0..1); Normalized String Type (page 349); A free-text value assigned as identifier of the property value. <p>The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See Flexible Properties (page 57))</p>																	
	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as a controlled or uncontrolled property value. 																	
Attribute(s)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
id (0..1)		XML Schema ID																
creator (0..1) creatoruri (0..1)		QCodeType IRIType																
modified (0..1)		DateOptTimeType																
custom (0..1)		XML Schema boolean																
pubconstraint (0..1) pubconstrainturi (0..1)		QCodeListType IRIListType																
how (0..1) howuri (0..1)		QCodeType IRIType																
why (0..1) whyuri (0..1)	QCodeType IRIType																	
	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>xml:lang (0..1)</td> <td>XML Schema language</td> </tr> <tr> <td>dir (0..1)</td> <td>XML Schema string: enumeration <i>ltr, rtl</i>.</td> </tr> </tbody> </table>	Name	Datatype	xml:lang (0..1)	XML Schema language	dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .											
Name	Datatype																	
xml:lang (0..1)	XML Schema language																	
dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .																	



Table 258. Flexible Property 2 Type

Child Element(s)	<ul style="list-style-type: none">▪ name (page 260) (0..unbounded)▪ hierarchyInfo (page 183) (0..unbounded)▪ sameAs {Relationship} (page 284)
XML Schema Note(s)	
Example(s)	



14.8.24 Integer 0 to 100 Type

Table 259. 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)	



14.8.25 Integer 1 to 9 Type

Table 260. 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)	



14.8.26 International String Type

Table 261. 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 360)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
	▪ commonPowerAttributes (page 361)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
why (0..1) whyuri (0..1)	QCodeType IRIType		
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.8.27 IRI List Type

Table 262. IRI List Type

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



14.8.28 IRI Type

Table 263. 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)	



14.8.29 Label 1 Type

Table 264. 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)	<ul style="list-style-type: none"> role (0..1); QCodeListType (page 349); roleuri (0..1); IRIListType (page 343); 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. 		
	<ul style="list-style-type: none"> commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
		why (0..1) whyuri (0..1)	QCodeType IRIType
	<ul style="list-style-type: none"> i18nAttributes (page 360) 	Name	Datatype
xml:lang (0..1)		XML Schema language	
dir (0..1)		XML Schema string: enumeration <i>ltr</i> , <i>rtl</i> .	
Child Element(s)	<ul style="list-style-type: none"> a (page 76) (0..unbounded) span (page 294) (0..unbounded) ruby (page 280) (0..unbounded) inline (page 189) (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)		



14.8.30 Link 1 Type

Table 265. 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	



Table 265. Link 1 Type (Continued)

Attribute(s)	<ul style="list-style-type: none"> rank (0..1); XML Schema nonNegativeInteger; The rank of the link among other links. 		
	<ul style="list-style-type: none"> rel (0..1); QCodeType (page 351); reluri (0..1); IRI Type (page 344); The identifier of the relationship between the current Item and the target resource. 		
	<ul style="list-style-type: none"> commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
	why (0..1) whyuri (0..1)	QCodeType IRIType	
	<ul style="list-style-type: none"> i18nAttributes (page 360) 	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
	<ul style="list-style-type: none"> timeValidityAttributes (page 361) 	Name	Data Type
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
	<ul style="list-style-type: none"> targetResourceAttributes (page 365) 	Name	Datatype
		href (0..1)	IRIType
		residref (0..1)	XML Schema string
		version (0..1)	XML Schema positiveInteger
contenttype (0..1)		XML Schema string	
contenttypevariant (0..1)		XML Schema string	
format (0..1) formaturi (0..1)		QCodeType IRIType	
size (0..1)	XML Schema non NegativeInteger		
	title (0..1)	XML Schema string	
<ul style="list-style-type: none"> guidref (0..1); XML Schema string; The use of this attribute is DEPRECATED, use <i>residref</i> instead. 			
<ul style="list-style-type: none"> hreftype (0..1); XML Schema normalizedString; The use of this attribute is DEPRECATED, use <i>contenttype</i> instead. 			



Table 265. Link 1 Type (Continued)

Child Element(s)	<ul style="list-style-type: none"> ▪ Hint and Extension Point (0..unbounded). Properties from the NAR namespace or from another XML namespace may be added. <p>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:</p> <ul style="list-style-type: none"> - 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)	<p>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.</p>
Example(s)	



14.8.31 Normalized String Type

Table 266. Normalized String Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	g2normalizedStringType
Title	Normalized string for G2
Definition	A string with a limited lexical space.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema string type with the restriction pattern [\S]* (note: includes a space character to the right of S!)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

14.8.32 QCode List Type

Table 267. 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)	



14.8.33 QCode Property Type

Table 268. 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)	<ul style="list-style-type: none"> ▪ qcode (0..1); QCodeType (page 351); A qualified code assigned as a property value. ▪ uri (0..1); XML Schema anyURI; A URI which identifies the same concept as qcode The use of either the qcode or the uri attribute is mandatory (from NewsML-G2 v 2.19 onwards) 																	
	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	<table border="1"> <thead> <tr> <th>Name</th> <th>Datatype</th> </tr> </thead> <tbody> <tr> <td>id (0..1)</td> <td>XML Schema ID</td> </tr> <tr> <td>creator (0..1) creatoruri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>modified (0..1)</td> <td>DateOptTimeType</td> </tr> <tr> <td>custom (0..1)</td> <td>XML Schema boolean</td> </tr> <tr> <td>pubconstraint (0..1) pubconstrainturi (0..1)</td> <td>QCodeListType IRIListType</td> </tr> <tr> <td>how (0..1) howuri (0..1)</td> <td>QCodeType IRIType</td> </tr> <tr> <td>why (0..1) whyuri (0..1)</td> <td>QCodeType IRIType</td> </tr> </tbody> </table>	Name	Datatype	id (0..1)	XML Schema ID	creator (0..1) creatoruri (0..1)	QCodeType IRIType	modified (0..1)	DateOptTimeType	custom (0..1)	XML Schema boolean	pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType	how (0..1) howuri (0..1)	QCodeType IRIType	why (0..1) whyuri (0..1)	QCodeType IRIType
		Name	Datatype															
		id (0..1)	XML Schema ID															
		creator (0..1) creatoruri (0..1)	QCodeType IRIType															
		modified (0..1)	DateOptTimeType															
		custom (0..1)	XML Schema boolean															
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType															
how (0..1) howuri (0..1)	QCodeType IRIType																	
why (0..1) whyuri (0..1)	QCodeType IRIType																	
Child Element(s)																		
XML Schema Note(s)																		
Example(s)																		



14.8.34 QCode Type

Table 269. 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)	



14.8.35 Qualified Property Type

Table 270. 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 350)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ i18nAttributes (page 360)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration <i>ltr, rtl</i> .
Child Element(s)	<ul style="list-style-type: none"> ▪ name (page 100) (0..unbounded) ▪ hierarchyInfo (page 183) (0..unbounded) 		
XML Schema Note(s)			
Example(s)			



14.8.36 Recurrence Rule Type

Table 271. 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)	<ul style="list-style-type: none"> ▪ 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 318); 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)	



14.8.37 Related Concept Type

Table 272. 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 <i>sameAs</i> , <i>broader</i> , or <i>narrower</i> .		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	Extends FlexPropType (page 336)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	<ul style="list-style-type: none"> rel (0..1); QCodeType (page 351); reluri (0..1); IRI Type (page 344); 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. 		
	<ul style="list-style-type: none"> timeValidityAttributes (page 361) 	Name	Data Type
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)	<ul style="list-style-type: none"> facet (page 165) (0..unbounded) related (0..unbounded) Note: has the same structure as related (page 267) except <related> as nested child element to limit the levels of recursion of <related> to a single one. sameAs {Relationship} (page 284) 		
XML Schema Note(s)			
Example(s)			



14.8.38 Rights Label Type

Table 273. 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 313)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ href (0..1); IRIType (page 344); The locator of a remote expression of rights.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.8.39 Truncated Date and Time Property Type

Table 274. 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)	<ul style="list-style-type: none"> ▪ commonPowerAttributes (page 361) 	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1) creatoruri (0..1)	QCodeType IRIType
		modified (0..1)	DateOptTimeType
		custom (0..1)	XML Schema boolean
		pubconstraint (0..1) pubconstrainturi (0..1)	QCodeListType IRIListType
		how (0..1) howuri (0..1)	QCodeType IRIType
	why (0..1) whyuri (0..1)	QCodeType IRIType	
Child Element(s)			
XML Schema Note(s)			
Example(s)			



14.8.40 Truncated Date and Time Type

Table 275. *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 may 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)	



14.8.41 Typed Qualified Property Type

Table 276. *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 352)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	<ul style="list-style-type: none"> ▪ type (0..1); QCodeType (page 351); typeuri (0..1); IRIType (page 344); The type of the concept assigned as property value.
Child Element(s)	
XML Schema Note(s)	
Example(s)	



14.8.42 Versioned String Type

Table 277. 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 342)
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)	



14.9 Attribute Group Definitions

14.9.1 Internationalization Attributes Group

Table 278. *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 <i>ltr, rtl</i> .	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.

14.9.2 Common Power Attributes Group



These attributes are used only at PCL.

Table 279. *commonPowerAttributes*

Title	Name	Card	Datatype	Definition
Local Identifier	id	0..1	XML Schema ID	The local identifier of the element.
Creator	creator creatoruri	0..1 0..1	QCodeType (page 351) IRI Type (page 344)	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 (page 318)	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.
Custom Property	custom	0..1	XML Schema boolean	If set to true the corresponding property was added to the G2 Item for a specific customer or group of customers only. The default value of this property is false which applies when this attribute is not used with the property.
Publication Constraint	pubcon- straint pubcon- strainturi	0..1 0..1	QCode List Type (page 349) IRI List Type (page 343)	One or many constraints that apply to publishing the value of the property. Each constraint applies to all descendant elements.
How	how howuri	0..1 0..1	QCodeType (page 351) IRI Type (page 344)	Indicates by which means the value was extracted from the content.
Why	why whyuri	0..1	QCodeType (page 351) IRI Type (page 344)	Why the metadata has been included.

In NewsML-G2 versions prior to 2.11 existed an Editing Attributes Group with id, creator and modified as member. In version 2.11 these attributes were transferred to this new group and the Editing Attributes Group was closed.

14.9.3 Time Validity Attributes Group

These attributes are used only at PCL.

Table 280. *timeValidityAttributes*

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



14.9.4 Ranking Attributes Group

These attributes are used only at PCL.

Table 281. *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"/>
```



14.9.5 Qualify Attributes Group

Table 282. *qualifyAttributes*

Title	Name	Card	Datatype	Definition
QCode	qcode	0..1	QCodeType (page 351)	A qualified code assigned as identifier of the property value.
URI	uri	0..1	XML Schema anyURI	A URI assigned as identifier of the property value.
Literal	literal	0..1	XML Schema normalizedString	A free-text value assigned as identifier of the property value.
Type	type	0..1	QCodeType (page 351)	The type of the concept assigned as a controlled or an uncontrolled property value.
	typeuri	0..1	IRI Type (page 344)	
Role	role	0..1	QCodeType (page 351)	A refinement of the semantics of the property.
	roleuri	0..1	IRI Type (page 344)	

Rule for @qcode and @uri in a property: (see **Qualified and Typed Properties** (page 56))

- An element SHOULD NOT use both a @qcode and a @uri.
- If both attributes, @qcode and @uri, are present the @qcode takes precedence.

Rule for @qcode/@uri and @literal in an element:

The use of qcode/uri and literal is mutually exclusive, only one of them or none is used. (See **Flexible Properties** (page 57))

14.9.6 Quantify Attributes Group

These attributes are used only at PCL.

Table 283. *quantifyAttributes*

Title	Name	Card	Datatype	Definition
Confidence	confidence	0..1	Int100Type (page 340)	The confidence with which the metadata has been assigned.
Relevance	relevance	0..1	Int100Type (page 340)	The relevance of the metadata to the news content to which it is attached.
Why	why	0..1	QCodeType (page 351)	Why the metadata has been included.
Derived From	derivedfrom	0..1	QCodeType (page 351)	A reference to the concept from which the concept identified by qcode was derived/ inferred - use DEPRECATED in NewsML-G2 2.12 and higher, use the derivedFrom element

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).

14.9.7 News Content Attributes

Table 284. *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.
	renditionuri	0..1	IRIType	
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.
Has Content	hascontent	0..1	XML Schema boolean	Indicates if the digital data of this rendition is available or not.

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 *contenttype* and *itemClass* in **Item Metadata** (page 203) 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”.



14.9.8 Target Resource Attributes Group

Table 285. *targetResourceAttributes*

Title	Name	Card	Data Type	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
Content Type Variant	contentypevariant	0..1	XML Schema string	A refinement of a generic content type (i.e. IANA MIME type) by a literal string value.
Format	format	0..1	QCodeType	A refinement of a generic content type (i.e. IANA MIME type) by a value from a controlled vocabulary.
	formaturi	0..1	IRIType	
Size	size	0..1	XML Schema non NegativeInteger	The size in bytes of the target resource
Title	title	0..1	XML Schema string	A short natural language name for the targeted resource.

14.9.9 News Content Characteristics

To be implemented as an attribute group.

Table 286. *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.
Line Count	linecount	0..1	XML Schema nonNegativeInteger	The count of lines of textual content. Applies to textual content.
Page Count	pagecount	0..1	XML Schema nonNegativeInteger	The count of pages of 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.
	widthunituri	0..1	IRIType	
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.
	heightunituri	0..1	IRIType	

Table 286. newsContentCharacteristics (Continued)

Title	Name	Card	Datatype	Definition
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 288 .
Image Layout Orientation	layoutorientation	0..1	QCodeType	Indicates whether the human interpretation of the top of the image is aligned to its short or long side. The recommended vocabulary is the IPTC Layout Orientation NewsCodes http://cv.iptc.org/newscodes/layoutorientation/
	layoutorientationuri	0..1	IRIType	
Image Colour Space	colourspace	0..1	QCodeType	The colour space of an image. Applies to image content.
	colourspaceuri	0..1	IRIType	
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/
	colourindicatoruri	0..1	IRIType	
Colour Depth	colourdepth	0..1	XML Schema nonNegativeInteger	The bit depth defining the spread of colour data within each sample.
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.
	durationunituri	0..1	IRIType	
Audio Codec	audiocodec	0..1	QCodeType	The applicable codec for audio data. Applies to audio content.
	audiocodec uri	0..1	IRIType	
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.

Table 286. *newsContentCharacteristics (Continued)*

Title	Name	Card	Datatype	Definition
Audio Channels	audiochannels	0..1	QCodeType	The audio sound system, e.g. <i>mono, stereo, surround</i> . Codes/URLs may represent e.g. <i>mono, stereo, surround</i> . Applies to audio content.
	audiochannelsuri	0..1	IRIType	
Video Codec	videocodec	0..1	QCodeType	The applicable codec for video data. Applies to video content.
	videocodecuri	0..1	IRIType	
Video Average Bit Rate	videoavgbitrate	0..1	XML Schema positiveInteger	The video bit rate in bps. Indicates the average variable bit rate if <i>videovbr</i> 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.
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 normalizedString	The video aspect ratio, e.g. 4:3 or 16:9. Applies to video content.
Video Sampling Method	videosampling	0..1	XML Schema normalizedString	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/
	videoscalinguri	0..1	IRIType	
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/
	videodefinitionuri	0..1	IRIType	

Table 287 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 287. *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 288 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. Orientation 1 is considered as default value.

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 hint describes how a picture of this orientation has to be flipped and/or rotated to show as the default orientation 1.

The column "Visual example" shows a picture of the character F having an orientation aligning with the value. The letters T(op), L(eft), R(ight) and B(ottom) represent the visual alignment of the image with orientation 1.

Table 288. 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. Hint: no action required.	
2	The 0th row is at the visual top of the image, and the 0th column is the visual right-hand side. Hint: flip horizontal.	
3	The 0th row is at the visual bottom of the image, and the 0th column is the visual right-hand side. Hint: rotate 180 degrees.	
4	The 0th row is at the visual bottom of the image, and the 0th column is the visual left-hand side. Hint: flip horizontal and rotate 180 degrees.	
5	The 0th row is the visual left-hand side of the image, and the 0th column is the visual top. Hint: flip vertical and rotate 90 degrees clockwise.	
6	The 0th row is the visual right-hand side of the image, and the 0th column is the visual top. Hint: rotate 90 degrees counterclockwise.	
7	The 0th row is the visual right-hand side of the image, and the 0th column is the visual bottom. Hint: flip vertical and rotate 90 degrees counterclockwise.	
8	The 0th row is the visual left-hand side of the image, and the 0th column is the visual bottom. Hint: rotate 90 degrees clockwise.	

15 Glossary

Table 289. Glossary

Term	Definition
alias	See scheme alias .
anonymous controlled vocabulary	A controlled vocabulary that is not a scheme .
catalog	A document 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 <i>subject</i> 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 289. 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 Meta-data . 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 289. 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 284) 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 .



16 References

Table 290. 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-G2/2.21/specification/
NewsML-G2	Specifications for NewsML-G2: http://www.iptc.org/std/NewsML-G2/2.21/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	
RFC2119	Key words for use in RFCs to Indicate Requirement Levels http://www.ietf.org/rfc/rfc2119.txt
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/xmlsig-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

17 Index

17.1 Elements

A

a ... 76
 access ... 67
 accessStatus ... 68
 accountable ... 69
 action ... 70
 address ... 255
 address {POI} ... 256
 affiliation ... 72
 altId ... 73
 altLoc ... 74
 altRep ... 75
 area ... 124
 assert ... 77
 assignedTo ... 79
 audience ... 80

B

bag ... 81
 bit ... 82
 born ... 134
 br ... 213
 broader ... 83
 by ... 84

C

capacity ... 85
 catalog ... 86
 catalogContainer ... 87
 catalogItem ... 88
 catalogRef ... 270
 ceasedToExist {Object} ... 127
 ceasedToExist {POI} ... 128
 channel {News Item} ... 89
 channel {News Message} ... 92
 circle ... 93
 concept ... 94
 conceptExtProperty ... 97
 conceptId ... 98
 conceptItem ... 99
 conceptRef ... 102
 conceptSet ... 103
 confirmation ... 104
 contactInfo ... 105
 contentCreated ... 129
 contentMeta (Concept Items) ... 115
 contentMeta (Knowledge Items) ... 109
 contentMeta (News Items) ... 111

contentMeta (Package Items) ... 113
 contentMeta {Catalog} ... 106
 contentMetaExtProperty ... 117
 contentModified ... 130
 contentSet ... 119
 contributor ... 120
 copyrightHolder ... 121
 copyrightNotice ... 122
 country ... 123
 created {Object} ... 135
 created {POI} ... 136
 created {Resource} ... 145
 creator ... 125
 creditline ... 126

D

dateline ... 143
 dates ... 144
 definition ... 96
 deliverableOf ... 147
 deliveredItemRef ... 148
 delivery ... 149
 derivedFrom ... 150
 description ... 146
 destination ... 151
 details ... 218
 died ... 137
 dissolved {Geo Area} ... 138
 dissolved {Organisation} ... 139
 duration ... 152

E

edNote ... 153
 email ... 155
 embargoed ... 131
 end ... 156
 event ... 157
 eventDetails ... 158
 events ... 160
 exclAudience ... 161
 exDate ... 162
 expires ... 164
 exRule ... 163

F

facet ... 165
 fax ... 166
 filename ... 167



firstCreated ... 132
 founded {Geo Area} ... 140
 founded {Organisation} ... 141

G

g2contentType ... 168
 generator ... 170
 genre ... 171
 geoAreaDetails ... 173
 group ... 174
 groupExtProperty ... 176
 groupRef ... 177
 groupSet ... 178

H

hash ... 179
 hasInstrument ... 180
 header ... 220
 headerExtProperty ... 221
 headline ... 182
 hierarchyInfo ... 183
 hop ... 184
 hopHistory ... 185

I

icon ... 186
 im ... 199
 incomingFeedId ... 189
 infoSource ... 200
 inline ... 189
 inlineData ... 191
 inlineRef ... 194
 inlineXML ... 195
 instanceOf ... 198
 itemClass ... 169, ... 201
 itemCount ... 202
 itemMeta ... 203
 itemMetaExtProperty ... 205
 itemRef ... 206
 itemSet ... 207

K

keyword ... 209
 knowledgeItem ... 210

L

language ... 211
 line {address} ... 71
 line {geoArea} ... 212
 link ... 214
 locality ... 215
 located ... 216
 location ... 217

location (EventsML-G2) ... 159

M

memberOf ... 219

N

name (Concept Name) ... 100
 name (Property Value Name) ... 260
 narrower ... 222
 newsContentCharacteristics ... 223
 newsCoverage ... 226
 newsCoverageExtProperty ... 229
 newsCoverageSet ... 230
 newsCoverageStatus ... 231
 newsItem (NewsML-G2) ... 232
 newsMessage ... 233
 note ... 101

O

objectDetails ... 234
 occurStatus ... 235
 openHours ... 236
 organisationDetails ... 238
 organiser ... 239
 origin ... 237
 origRep ... 240

P

packageItem ... 241
 participant ... 242
 participationRequirement ... 243
 partMeta ... 244
 partMetaExtProperty ... 246
 party ... 247
 personDetails ... 248
 phone ... 249
 planning ... 250
 planningExtProperty ... 251
 planningItem ... 252
 POIDetails ... 253
 polygon ... 254
 position ... 172
 postalCode ... 257
 priority ... 258
 profile ... 259
 provider ... 118
 pubStatus ... 261

R

rating ... 262
 rb ... 281
 rDate ... 264
 regionDelim ... 305



registration ... 266
related ... 267
remoteContent ... 271
remoteContentExtProperty ... 274
remoteInfo ... 275
rightsInfo ... 276
rightsInfoExtProperty ... 277
role ... 278, ... 279
rp ... 282
rRule ... 265
rt ... 283
ruby ... 280

S

sameAs ... 284
sameAsScheme ... 286
scheduled ... 287
scheme ... 288
schemeMeta ... 289
schemeMetaExtProperty ... 290
sender ... 291
sent ... 142
service ... 154
signal ... 292
slugline ... 293
span ... 294
start ... 295
subject ... 296

T

timeDelim ... 297
timestamp ... 299
title (Item Title) ... 208
transmitId ... 300
type ... 301

U

urgency ... 303
usageTerms ... 304
userInteraction ... 302

V

versionCreated ... 133

W

web ... 306



17.2 Datatypes

A

AnyltemType ... 310

ApproximateDateTimePropType ... 311

AudienceType ... 312

B

BlockType ... 313

C

ConceptIdType ... 315

ContentMetadataType ... 316

D

DateOptTimePropType ... 317

DateOptTimeType ... 318

DateTimeOrNullPropType ... 319

DateTimePropType ... 320

E

ElectronicAddressTechType ... 322

ElectronicAddressType ... 321

F

Flex1ConceptPropType ... 323

Flex1ExtPropType ... 324

Flex1PartyPropType ... 324

Flex1PropType ... 326

Flex2ExtPropType ... 328

FlexLocationPropType ... 328

FlexOrganisationPropType ... 330

FlexPartyPropType ... 332

FlexPersonPropType ... 334

FlexProp2Type ... 338

FlexPropType ... 336

G

g2normalizedStringType ... 349

I

Int100Type ... 340

Int1To9Type ... 341

IntlStringType ... 342

IRIType ... 344

L

Label1Type ... 345

Link1Type ... 346

N

Normalized String Type ... 349

Q

QCodeListType ... 343, ... 349

QCodePropType ... 350

QCodeType ... 351

QualPropType ... 352

R

RecurrenceRuleType ... 353

RelatedConceptType ... 354

RightsLabelType ... 355

T

TruncatedDateTimePropType ... 356

TruncatedDateTimeType ... 357

TypedQualPropType ... 358

V

VersionedStringType ... 359



Contact Information

Contact the IPTC by:

Postal mail:
25 Southampton Buildings
London WC2A 1AL
United Kingdom

Web:

www.iptc.org

Twitter:

[@IPTC](https://twitter.com/IPTC)
[@IPTCupdates](https://twitter.com/IPTCupdates)

Email:

office@iptc.org