

Network Working Group  
Internet-Draft  
Obsoletes: 2629 (if approved)  
Intended status: Informational  
Expires: March 7, 2016

P. Hoffman  
ICANN  
September 4, 2015

The 'XML2RFC' version 3 Vocabulary  
draft-hoffman-xml2rfc-23

Abstract

This document defines the "XML2RFC" version 3 vocabulary; an XML-based language used for writing RFCs and Internet-Drafts. It is heavily derived from the version 2 vocabulary that is also under discussion. This document obsoletes the v2 grammar described in RFC 2629 and its expected followup, draft-iab-xml2rfc.

Editorial Note (To be removed by RFC Editor)

Discussion of this draft takes place on the rfc-interest mailing list (rfc-interest@rfc-editor.org), which has its home page at <<https://www.rfc-editor.org/mailman/listinfo/rfc-interest>>.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 7, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents

(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1.	Introduction . . . . .	5
1.1.	Design Criteria for the Changes in v3 . . . . .	5
1.2.	Differences from v2 to v3 . . . . .	6
1.2.1.	New Elements in v3 . . . . .	6
1.2.2.	New Attributes for Existing Elements . . . . .	7
1.2.3.	Elements and Attributes Deprecated from v2 . . . . .	8
1.2.4.	Additional Changes from v2 . . . . .	9
1.3.	Syntax Notation . . . . .	10
2.	Elements . . . . .	10
2.1.	<abstract> . . . . .	11
2.2.	<address> . . . . .	11
2.3.	<annotation> . . . . .	12
2.4.	<area> . . . . .	12
2.5.	<artwork> . . . . .	13
2.6.	<aside> . . . . .	16
2.7.	<author> . . . . .	17
2.8.	<back> . . . . .	18
2.9.	<bcp14> . . . . .	19
2.10.	<blockquote> . . . . .	19
2.11.	<boilerplate> . . . . .	21
2.12.	  . . . . .	21
2.13.	<city> . . . . .	21
2.14.	<code> . . . . .	21
2.15.	<country> . . . . .	22
2.16.	<cref> . . . . .	22
2.17.	<date> . . . . .	23
2.18.	<dd> . . . . .	24
2.19.	<displayreference> . . . . .	26
2.20.	<dl> . . . . .	26
2.21.	<dt> . . . . .	27
2.22.	<em> . . . . .	28
2.23.	<email> . . . . .	29
2.24.	<eref> . . . . .	29
2.25.	<figure> . . . . .	30
2.26.	<front> . . . . .	32
2.27.	<iref> . . . . .	33
2.28.	<keyword> . . . . .	34
2.29.	<li> . . . . .	34

2.30.	<link>	36
2.31.	<middle>	37
2.32.	<name>	37
2.33.	<note>	37
2.34.	<ol>	38
2.35.	<organization>	40
2.36.	<phone>	41
2.37.	<postal>	41
2.38.	<postalLine>	42
2.39.	<refcontent>	42
2.40.	<reference>	43
2.41.	<referencegroup>	44
2.42.	<references>	44
2.43.	<region>	45
2.44.	<relref>	45
2.45.	<rfc>	49
2.46.	<section>	53
2.47.	<seriesInfo>	55
2.48.	<sourcecode>	57
2.49.	<street>	59
2.50.	<strong>	59
2.51.	<sub>	60
2.52.	<sup>	61
2.53.	<t>	62
2.54.	<table>	64
2.55.	<tbody>	64
2.56.	<td>	65
2.57.	<tfoot>	67
2.58.	<th>	67
2.59.	<thead>	69
2.60.	<title>	69
2.61.	<tr>	70
2.62.	<tt>	70
2.63.	<ul>	71
2.64.	<uri>	72
2.65.	<workgroup>	72
2.66.	<xref>	72
2.67.	<svg> (in namespace <a href="http://www.w3.org/2000/svg">http://www.w3.org/2000/svg</a> )	75
3.	Elements from v2 That Have Been Deprecated	75
3.1.	<c>	75
3.2.	<facsimile>	76
3.3.	<format>	76
3.4.	<list>	76
3.5.	<postamble>	77
3.6.	<preamble>	78
3.7.	<spanx>	78
3.8.	<texttable>	79
3.9.	<ttable>	80

3.10. <vspace> . . . . .	81
4. Internationalization Considerations . . . . .	81
5. Security Considerations . . . . .	81
6. IANA Considerations . . . . .	82
6.1. Internet Media Type Registration . . . . .	82
6.2. Link Relation Registration . . . . .	83
7. Acknowledgments . . . . .	84
8. References . . . . .	84
8.1. Normative References . . . . .	84
8.2. Informative References . . . . .	84
Appendix A. Front Page ('Boilerplate') Generation . . . . .	88
A.1. The /rfc/@ipr Attribute . . . . .	88
A.1.1. Current Values: '*trust200902' . . . . .	88
A.1.2. Historic Values . . . . .	90
A.2. The /rfc/@submissionType Attribute . . . . .	90
A.3. The /rfc/@consensus Attribute . . . . .	91
Appendix B. The v3 Format and Processing Tools . . . . .	91
B.1. Including External Text with XInclude . . . . .	93
B.2. Anchors and IDs . . . . .	94
B.2.1. Overlapping Values . . . . .	94
B.3. Attributes Controlled by the Prep Tool . . . . .	95
Appendix C. Relax NG Schema . . . . .	97
Appendix D. Schema Differences from v2 . . . . .	118
Index . . . . .	138

## 1. Introduction

This document describes version 3 ("v3") of the "XML2RFC" vocabulary; an XML-based language ('Extensible Markup Language', [XML]) used for writing RFCs ([RFC7322]) and Internet-Drafts ([IDGUIDE]).

This document obsoletes the version 2 vocabulary ("v2") [XML2RFCv2], which contains the extended language definition. That document in turn obsoletes the original version ("v1") [RFC2629]. This document directly copies the material from [XML2RFCv2] where possible; as that document makes its way toward RFC publication, this document will incorporate as many of the changes as possible.

The v3 format will be used as part of the new RFC series described in [RFC6949]. The new format will be handled by one or more new tools for preparing the XML and converting it to other representations. Features of the expected tools are described in Appendix B. That section defines some terms used throughout this document, such as "prep tool" and "formatter".

Note that the vocabulary contains certain constructs that might not be used when generating of the final text; however, they can provide useful data for other uses (such as index generation, populating a keyword database, or syntax checks).

In this document, the term "format" is used when describing types of documents, primarily XML and HTML. The term "representation" is used when talking about a specific instantiation of a format, such as an XML document or an HTML document that was created by an XML document.

The details (particularly any vocabularies) described in this document are expected to change based on experience gained in implementing the RFC production center's toolset. Revised documents will be published capturing those changes as the toolset is completed. Other implementers must not expect those changes to remain backwards-compatible with the details described in this document.

The following two sections are a hopefully-complete list of all the technical changes between [XML2RFCv2] and this document, as well as the design criteria for those changes.

### 1.1. Design Criteria for the Changes in v3

The design criteria of the changes from v2 to v3 are:

- o The intention is that starting and editing a v3 document will be easier than for a v2 document.

- o There will be good v2-to-v3 conversion tools for when an author wants to change versions.
- o There are no current plans to make v3 XML the required submission representation for drafts or RFCs. That might happen eventually, but it is likely to be years away.

There is a desire to keep as much of the v2 grammar as makes sense within the above design criteria and not to make gratuitous changes to the v2 grammar. Another way to say this is "we would rather encourage backward compatibility but not be constrained by it". Still, the goal of starting and editing a v3 document being easier than for a v2 document is more important than backwards compatibility with v2, given the latter two design criteria.

v3 is upwards compatible with v2, meaning that a v2 document is meant to be a valid v3 document as well. However, some features of v2 are deprecated in v3 in favor of new elements. Deprecated features are listed in Section 1.2.3, and are described in [XML2RFCv2].

## 1.2. Differences from v2 to v3

The format changes in v3 are listed in the following subsections.

### 1.2.1. New Elements in v3

- o Add <dl>, <ul>, and <ol> as new ways to make lists. This is a significant change from v2 in that the child under these elements is <li>, not <t>. <li> has a model of either containing one or more <t> elements, or containing the flowing text normally found in <t>. These lists are children of <section>s and other lists instead of <t>.
- o Add <strong>, <em>, <tt>, <sub>, and <sup> for character formatting.
- o Add <aside> for incidental text that will be indented when displayed.
- o Add <sourcecode> to differentiate from <artwork>.
- o Add <table>, <thead>, <tbody>, <tfoot>, <tr>, <td>, and <th> to give table functionality like that in HTML.
- o Add <boilerplate> to hold the automatically-generated boilerplate text.

- o Add `<blockquote>` to indicate a quotation as in a paragraph-like format.
- o Add `<name>` to sections, notes, figures, and texttables to allow character formatting (fixed-width font) in their titles, and to allow references in the names.
- o Add `<postalLine>`, free text that represents one line of the address.
- o Add `<displayreference>` to allow display of more mnemonic anchor names for automatically-included references.
- o Add `<refcontent>` to allow better control of text in a reference.
- o Add `<referencegroup>` to allow referencing multi-RFC documents such as STDs and BCPs.
- o Add `<relref>` to allow referncing specific sections or anchors in references.
- o Add `<link>` to point to a resource related to the RFC.
- o Add `<br>` to allow line breaks (but not blank lines) in the generated output for table cells.
- o Add `<svg>` to allow easy inclusion of SVG drawings in `<artwork>`.

#### 1.2.2. New Attributes for Existing Elements

- o Add "sortRefs", "symRefs", "tocDepth", and "tocInclude" attributes to `<rfc>` to cover processor instructions (PIs) that were in v2 that are still needed in the grammar. Add "prepTime" to indicate the time that the XML went through a preparation step. Add "version" to indicate the version of XML2RFC vocabulary used in the document. Add "scripts" to indicate which scripts are needed to render the document. Add "expiresDate" when an Internet Draft expires.
- o Add "ascii" attributes to `<email>`, `<organization>`, `<street>`, `<city>`, `<region>`, `<country>`, and `<code>`. Also add "asciiFullname", "asciiInitials", and "asciiSurname" to `<author>`. This allows an author to specify their information in their native scripts as the primary entry and still allow the ASCII-equivalent values to appear in the processed documents.
- o Add "anchor" attributes to many block elements to allow them to be linked with `<relref>` and `<xref>`.

- o Add the "section", "relative", and "sectionFormat" attributes to <xref>.
- o Add the "numbered" and "removeInRFC" attributes to <section>.
- o Add the "removeInRFC" attribute to <note>.
- o Add "pn" to <artwork>, <aside>, <blockquote>, <boilerplate>, <dt>, <figure>, <li>, <section> <sourcecode>, <t>, and <table> to hold automatically generated numbers for items in a section that don't have their own numbering (namely figures and tables).
- o Add "display" to <cref> to indicate to tools whether or not to display the comment.
- o Add "keepWithNext" and "keepWithPrevious" to <t> as a hint to tools that do pagination that they should try to keep the paragraph with the next/previous element.

### 1.2.3. Elements and Attributes Deprecated from v2

Deprecated elements and attributes are legacy vocabulary from v2 that are supported for input to v3 tools. They are likely to be removed from those tools in the future. Instead of being listed in Section 2, they are listed in in Section 3. See Appendix B for more information on tools and how they will handle deprecated features.

- o Deprecate <list> in favor of <dl>, <ul>, and <ol>.
- o Deprecate <spanx>; replace it with <strong>, <em>, and <tt>.
- o Deprecate <vspace> because the major use for it, creating pseudo-paragraph-breaks in lists, is now handled properly.
- o Deprecate <texttable>, <ttable>, and <c>; replace them with the new table elements (<table> and the elements that can be contained within it).
- o Deprecate <facsimile> because it is rarely used and is not actually useful; <email> is a much more useful way to get in touch with authors.
- o Deprecate <format> because it is not useful and has caused surprise for authors in the past. If the goal is to provide a single URI (Uniform Resource Identifier) for a reference, use the "target" attribute on <reference> instead.



- o Deprecate <preamble> and <postamble> in favor of simply using <t> before or after the figure. This also deprecates the "align" attribute in <figure>.
- o Deprecate the "title" attribute in <section>, <note>, <figure>, <references>, and <texttable> in favor of the new <name>.
- o Deprecate the "alt", and "src" attributes in <figure> because they overlap with the attributes in <artwork>.
- o Deprecate the "xml:space" attribute in <artwork> because there was only one useful value. Deprecate "height" and "width" attribute in both <artwork> and <figure> because they are not needed for the new output formats.
- o Deprecate the "pageno" attribute in <xref> because it was unused in v2. Deprecate the "none" values for the "format" attribute in <xref> because it makes no sense semantically.

#### 1.2.4. Additional Changes from v2

- o Allow non-ASCII characters in the format; the characters that are actually allowed will be determined by the RFC Editor.
- o Allow <artwork> and <sourcecode> to be used on their own in <section> (no longer confine them to a figure).
- o Give more specifics of handling the "type" attribute in <artwork>.
- o Allow <strong>, <em>, <tt>, <eref>, and <xref> in <cref>.
- o Allow the sub-elements inside a <reference> to be in any order.
- o Turned off the auto-generation of anchors in <cref> because there is no use case for them that cannot be achieved in other ways.
- o Allow more than one <artwork>, or more than one <sourcecode>, in <figure>.
- o In <front>, make <date> optional.
- o In <postal>, allow the sub-elements to be in any order. Also allow the inclusion of the new <postalLine> instead of the older elements.
- o In <section>, restricted the names of the anchors that can be used on some types of sections.

- o Made <seriesInfo> a child of <front>, and deprecated it as a child of <reference>. This also deprecates some of the attributes from <rfc> and moves them into <seriesInfo>.
- o <t> now only contains non-block elements, so it no longer contains <figure> elements.
- o Do not generate the grammar from a DTD, but instead get it directly from the Relax Next Generation (RNG) grammar [RNG].

### 1.3. Syntax Notation

The XML vocabulary here is defined in prose, based on the Relax NG schema ([RNC]) contained in Appendix C (specified in Relax NG Compact Notation, "RNC").

Note that the schema can be used for automated validity checks, but certain constraints are only described in prose (example: the conditionally required presence of the "abbrev" attribute).

## 2. Elements

The sections below describe all elements and their attributes.

Note that attributes not labeled "mandatory" are optional.

Many elements have an optional "anchor" attribute. In all cases, the value of the "anchor" attribute needs to be a valid XML "Name" (Section 2.3 of [XML]), additionally constrained to US-ASCII characters ([USASCII]). Thus, the character repertoire consists of "A-Z", "a-z", "0-9", "\_", "-", ".", and ":", where "0-9", ".", and "-" are disallowed as start character. Anchors are described in more detail in Appendix B.2.

Tools interpreting the XML described here will collapse horizontal whitespace and linebreaks to a single whitespace (except inside <artwork> and <sourcecode>), and will trim leading and trailing whitespace.

Some of the elements have attributes that are not described in this section because those elements are specific to the prep tool. People writing tools to process this format should read all of the appendices for a complete description of these attributes.

Every element in the v3 vocabulary can have an "xml:lang" attribute, an "xml:base" attribute, or both. The xml:lang attribute specifies the language used in the element. This is sometimes useful for renderers which display different fonts for ideographic characters

used in China and Japan. The `xml:base` attribute is sometimes added to an XML file when doing XML-to-XML conversion where the base file has `XInclude` attributes (see Appendix B.1).

## 2.1. <abstract>

Contains the abstract of the document. See [RFC7322] for more information on restrictions for the abstract.

This element appears as a child element of: <front> (Section 2.26).

Content model:

In any order, but at least one of:

- o <dl> elements (Section 2.20)
- o <ol> elements (Section 2.34)
- o <t> elements (Section 2.53)
- o <ul> elements (Section 2.63)

### 2.1.1. 'anchor' attribute

Document-wide unique identifier for the abstract.

## 2.2. <address>

Provides address information for the author.

This element appears as a child element of: <author> (Section 2.7).

Content model:

In this order:

1. One optional <postal> element (Section 2.37)
2. One optional <phone> element (Section 2.36)
3. One optional <facsimile> element (Section 3.2)
4. One optional <email> element (Section 2.23)
5. One optional <uri> element (Section 2.64)

### 2.3. <annotation>

Provides additional prose augmenting a bibliographical reference. This text is intended to be shown after the rest of the generated reference text.

This element appears as a child element of: <reference> (Section 2.40).

Content model:

In any order:

- o Text
- o <bcp14> elements (Section 2.9)
- o <cref> elements (Section 2.16)
- o <em> elements (Section 2.22)
- o <eref> elements (Section 2.24)
- o <iref> elements (Section 2.27)
- o <relref> elements (Section 2.44)
- o <spanx> elements (Section 3.7)
- o <strong> elements (Section 2.50)
- o <sub> elements (Section 2.51)
- o <sup> elements (Section 2.52)
- o <tt> elements (Section 2.62)
- o <xref> elements (Section 2.66)

### 2.4. <area>

Provides information about the IETF area to which this document relates (currently not used when generating documents).

The value ought to be either the full name or the abbreviation of one of the IETF areas as listed on <<http://www.ietf.org/iesg/area.html>>. The list will be kept by the RFC Editor.

This element appears as a child element of: <front> (Section 2.26).

Content model: only text content.

## 2.5. <artwork>

This element allows the inclusion of "artwork" into the document. <artwork> provides full control of horizontal whitespace and line breaks, and thus is used for a variety of things, such as diagrams ("line art") and protocol unit diagrams.

Alternatively, the "src" attribute allows referencing an external graphics file, such as a vector drawing in SVG or a bitmap graphic file, using a URI. In this case, the textual content acts as fallback for output representations that do not support graphics, and thus ought to contain either a "line art" variant of the graphics, or otherwise prose that describes the included image in sufficient detail.

If the artwork includes either "&" or "<" characters, or the string "]]>" those characters need to be encoded using escaping or CDATA block(s); see <sourcecode> for a fuller description of these solutions.

In [XML2RFCv2], the <artwork> element was also used for source code and formal languages; in v3, this is now done with <sourcecode>.

There are at least five ways to include SVG in artwork in Internet Drafts:

- o Inline, by including all of the SVG in the content of the element, such as: <artwork type="svg"><svg xmlns...">
- o Inline, but using XInclude (see Appendix B.1), such as: <artwork type="svg"><xi:include href=...>
- o As a data: URI, such as: <artwork type="svg" src="data:image/svg+xml,%3Csvg%20xmlns%3D%22http%3A%2F%2Fwww.w3...">
- o As a URI to an external entity, such as: <artwork type="svg" src="http://www.example.com/...">
- o As a local file, such as: <artwork type="svg" src="diagram12.svg">

The use of SVG in Internet Drafts and RFCs is covered in much more detail in [SVGforRFCs].

The above methods for inclusion of SVG art can also be used for

including text artwork, but using a data: URI is probably confusing for text artwork.

Formatters that do pagination should attempt to keep artwork on a single page. This is to prevent artwork that is split across pages from looking like two separate pieces of artwork.

This element appears as a child element of: <aside> (Section 2.6), <blockquote> (Section 2.10), <dd> (Section 2.18), <figure> (Section 2.25), <li> (Section 2.29), <section> (Section 2.46), <td> (Section 2.56), and <th> (Section 2.58).

Content model:

Either:

Text

Or:

<svg> elements (Section 2.67)

#### 2.5.1. 'align' attribute

Controls whether the artwork appears left justified (default), centered, or right justified.

Allowed values:

- o "left" (default)
- o "center"
- o "right"

#### 2.5.2. 'alt' attribute

Alternative text description of the artwork (which is more than just a summary or caption). When the art comes from the "src" attribute, and the format of that artwork supports alternate text, the alternative text comes from the text of the artwork itself, not from this attribute. The contents of this attribute are important to readers who are visually impaired, as well as those reading on devices that cannot show the artwork well, or at all.

#### 2.5.3. 'anchor' attribute

Document-wide unique identifier for this artwork.

#### 2.5.4. 'height' attribute

Deprecated.

#### 2.5.5. 'name' attribute

A filename suitable for the contents (such as for extraction to a local file). This attribute can be helpful for other kinds of tools (such as automated syntax checkers which work by extracting the artwork). Note that the "name" attribute does not need to be unique for artwork elements in a document. If multiple artwork elements have the same name attribute, a processing tool might assume that the elements are all fragments of a single file, and the tool can collect those fragments for later processing. See Section 5 for a discussion of possible problems with the value of this attribute.

#### 2.5.6. 'src' attribute

The URI reference of a graphics file ([RFC3986]), or the name of a file on the local disk. This can be a "data" URI ([RFC2397]) that contains the contents of the graphics file. Note that the inclusion of art with the "src" attribute depends on the capabilities of the processing tool reading the XML document. Tools need to be able to handle the file: URI, and should be able to handle http: and https: URIs as well. The prep tool will be able to handle reading the "src" attribute.

If no URI scheme is given in the attribute, the attribute is considered to be a local file name. Processing tools must be careful to not accept dangerous values for the filename, particularly those that contain absolute references outside the current directory.

In some cases, the prep tool may remove the "src" attribute after processing its value. See [PREPTOOL] for a description of this.

It is an error to have both a "src" attribute and content in the <artwork> element.

#### 2.5.7. 'type' attribute

Specifies the type of the artwork. The value of this attribute is free text with certain values designated as preferred.

The preferred values for <artwork> types are:

- o `ascii-art`
- o `binary-art`
- o `call-flow`
- o `hex-dump`
- o `svg`

The RFC Editor will maintain a complete list of the preferred values on its web site, and that list is expected to be updated over time. Thus, a consumer of v3 XML should not cause a failure when it encounters an unexpected type.

#### 2.5.8. `'width'` attribute

Deprecated.

#### 2.5.9. `'xml:space'` attribute

Deprecated.

#### 2.6. `<aside>`

This element is a container for content that is semantically less important or tangential to the content that surrounds it.

This element appears as a child element of: `<section>` (Section 2.46).

Content model:

In any order:

- o `<artwork>` elements (Section 2.5)
- o `<dl>` elements (Section 2.20)
- o `<figure>` elements (Section 2.25)
- o `<iref>` elements (Section 2.27)
- o `<list>` elements (Section 3.4)
- o `<ol>` elements (Section 2.34)
- o `<t>` elements (Section 2.53)



- o <table> elements (Section 2.54)
- o <ul> elements (Section 2.63)

#### 2.6.1. 'anchor' attribute

Document-wide unique identifier for this aside.

#### 2.7. <author>

Provides information about a document's author. This is used both for the document itself (at the beginning of the document) and for referenced documents.

The <author> elements contained within the document's <front> element are used to fill the boilerplate, and also to generate the "Author's Address" section (see [RFC7322]).

Note that an "author" can also be just an organization (by not specifying any of the name attributes, but adding the <organization> child element).

Furthermore, the "role" attribute can be used to mark an author as "editor". This is reflected both on the front page and in bibliographical references. Note that this specification does not define a precise meaning for the term "editor".

See Section "Authors vs. Contributors" of [RFCPOLICY] for more information.

This element appears as a child element of: <front> (Section 2.26).

Content model:

In this order:

1. One optional <organization> element (Section 2.35)
2. One optional <address> element (Section 2.2)

##### 2.7.1. 'asciiFullname' attribute

The ASCII equivalent of the author's full name.

##### 2.7.2. 'asciiInitials' attribute

The ASCII equivalent of the author's initials.

#### 2.7.3. 'asciiSurname' attribute

The ASCII equivalent of the author's surname.

#### 2.7.4. 'fullname' attribute

The full name (used in the automatically generated "Author's Address" section).

#### 2.7.5. 'initials' attribute

Author initials (used on the front page and in references).

The value contains one or more initials, each followed by a period. Initials should be provided as a whitespace separated list of pairs of a letter and a dot.

#### 2.7.6. 'role' attribute

Specifies the role the author had in creating the document.

Allowed values:

- o "editor"

#### 2.7.7. 'surname' attribute

The author's surname (used on the front page and in references).

### 2.8. <back>

Contains the "back" part of the document: the references and appendices. In <back>, <section> elements indicate appendices.

This element appears as a child element of: <rfc> (Section 2.45).

Content model:

In this order:

1. Optional <displayreference> elements (Section 2.19)
2. Optional <references> elements (Section 2.42)
3. Optional <section> elements (Section 2.46)

## 2.9. <bcp14>

Marks text that are phrases defined in BCP 14 such as "MUST", "SHOULD NOT", and so on. When shown in some of the output representations, the text in this element might be highlighted. The use of this element is optional.

This element is only to be used around the actual phrase from BCP 14, not the full definition of a requirement. For example, it is correct to say "The packet <bcp14>MUST</bcp14> be dropped.", but it not correct to say "<bcp14>The packet MUST be dropped.</bcp14>".

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <preamble> (Section 3.6), <refcontent> (Section 2.39), <strong> (Section 2.50), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), and <tt> (Section 2.62).

Content model: only text content.

## 2.10. <blockquote>

Specifies a block of text is a quotation.

This element appears as a child element of: <section> (Section 2.46).

Content model:

Either:

In any order, but at least one of:

- \* <artwork> elements (Section 2.5)
- \* <dl> elements (Section 2.20)
- \* <figure> elements (Section 2.25)
- \* <ol> elements (Section 2.34)
- \* <sourcecode> elements (Section 2.48)
- \* <t> elements (Section 2.53)
- \* <ul> elements (Section 2.63)

Or:

In any order, but at least one of:

- \* Text
- \* `<bcpl4>` elements (Section 2.9)
- \* `<cref>` elements (Section 2.16)
- \* `<em>` elements (Section 2.22)
- \* `<eref>` elements (Section 2.24)
- \* `<iref>` elements (Section 2.27)
- \* `<relref>` elements (Section 2.44)
- \* `<strong>` elements (Section 2.50)
- \* `<sub>` elements (Section 2.51)
- \* `<sup>` elements (Section 2.52)
- \* `<tt>` elements (Section 2.62)
- \* `<xref>` elements (Section 2.66)

#### 2.10.1. 'anchor' attribute

Document-wide unique identifier for this quotation.

#### 2.10.2. 'cite' attribute

The source of the citation. This must be a URI. If the `quotedFrom` attribute is given, this URI will be used by processing tools as the link for the text of that attribute.

#### 2.10.3. 'quotedFrom' attribute

Name of person or document the text in this element is quoted from. A formatter should render this as visible text at the end of the quotation.

### 2.11. <boilerplate>

Holds the boilerplate text for the document. This section is filled in by the prep tool.

This element appears as a child element of: <front> (Section 2.26).

Content model:

One or more <section> elements (Section 2.46)

### 2.12. <br>

Indicates that a line break should be inserted in the generated output by a formatting tool. Multiple successive instances of this element do not cause blank lines to appear in the output, and is thus not useful.

This element appears as a child element of: <td> (Section 2.56), and <th> (Section 2.58).

Content model: this element does not have any contents.

### 2.13. <city>

Gives the city name in a postal address.

This element appears as a child element of: <postal> (Section 2.37).

Content model: only text content.

#### 2.13.1. 'ascii' attribute

The ASCII equivalent of the city name.

### 2.14. <code>

Gives the postal region code.

This element appears as a child element of: <postal> (Section 2.37).

Content model: only text content.

#### 2.14.1. 'ascii' attribute

The ASCII equivalent of the postal code.

## 2.15. <country>

Gives the country name or code in a postal address.

This element appears as a child element of: <postal> (Section 2.37).

Content model: only text content.

### 2.15.1. 'ascii' attribute

The ASCII equivalent of the country name.

## 2.16. <cref>

Represents a comment.

Comments can be used in a document while it is work-in-progress. They might appear either inline and visually highlighted, at the end of the document, or not at all, depending on the formatting tool.

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <c> (Section 3.1), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <name> (Section 2.32), <postamble> (Section 3.5), <preamble> (Section 3.6), <strong> (Section 2.50), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), <tt> (Section 2.62), and <ttcol> (Section 3.9).

Content model:

In any order:

- o Text
- o <em> elements (Section 2.22)
- o <eref> elements (Section 2.24)
- o <relref> elements (Section 2.44)
- o <strong> elements (Section 2.50)
- o <sub> elements (Section 2.51)
- o <sup> elements (Section 2.52)

- o `<tt>` elements (Section 2.62)
- o `<xref>` elements (Section 2.66)

#### 2.16.1. 'anchor' attribute

Document-wide unique identifier for this comment.

#### 2.16.2. 'display' attribute

Suggests whether or not the the comment should be displayed by formatting tools. This might be set to "false" if you want to keep a comment in a document after the contents of the comment have already been dealt with.

Allowed values:

- o "true" (default)
- o "false"

#### 2.16.3. 'source' attribute

Holds the "source" of a comment, such as the name or the initials of the person who made the comment.

#### 2.17. `<date>`

Provides information about the publication date.

Note that this element is used both for the boilerplate of the document being produced, and also inside bibliographic references that use the `<front>` element.

In the boilerplate case, it defines the date of publication for the current document (Internet Draft or RFC). When producing Internet-Drafts, the prep tool uses this date to compute the expiration date (see [IDGUIDE]). When one or more of "year", "month", or "day" are left out, the prep tool will attempt to use the current system date if the attributes that are present are consistent with that date.

Also in the first case, that month names, if given, need to match the full English month name: "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", or "December".

When the prep tool is used to create Internet Drafts, it will reject a submitted Internet Draft that has a `<date>` element in the

boilerplate for itself that is anything other than today. That is, the tool will not allow a submitter to specify a date other than the day of submission. To avoid this problem, authors might simply not include a <date> element in the boilerplate.

In the case of bibliographic references, the date information can have prose text for the month or year. For example, vague dates (year="ca. 2000"), date ranges (year="2012-2013"), non-specific months (month="Second quarter") and so on, are allowed.

This element appears as a child element of: <front> (Section 2.26).

Content model: this element does not have any contents.

#### 2.17.1. 'day' attribute

In the "boilerplate" case: the day of publication; this is a number. Otherwise: an indication of the publication day, with the format not being restricted.

#### 2.17.2. 'month' attribute

In the "boilerplate" case: the month of publication; this is the English name of the month. Otherwise: an indication of the publication month, with the format not being restricted.

#### 2.17.3. 'year' attribute

In the "boilerplate" case: the year of publication; this is a number (usually four-digit). Otherwise: an indication of the publication year, with the format not being restricted.

### 2.18. <dd>

The definition part of an entry in a definition list.

This element appears as a child element of: <dl> (Section 2.20).

Content model:

Either:

In any order, but at least one of:

\* <artwork> elements (Section 2.5)



- \* <dl> elements (Section 2.20)
- \* <figure> elements (Section 2.25)
- \* <ol> elements (Section 2.34)
- \* <sourcecode> elements (Section 2.48)
- \* <t> elements (Section 2.53)
- \* <ul> elements (Section 2.63)

Or:

In any order, but at least one of:

- \* Text
- \* <bcp14> elements (Section 2.9)
- \* <cref> elements (Section 2.16)
- \* <em> elements (Section 2.22)
- \* <eref> elements (Section 2.24)
- \* <iref> elements (Section 2.27)
- \* <relref> elements (Section 2.44)
- \* <strong> elements (Section 2.50)
- \* <sub> elements (Section 2.51)
- \* <sup> elements (Section 2.52)
- \* <tt> elements (Section 2.62)
- \* <xref> elements (Section 2.66)

#### 2.18.1. 'anchor' attribute

Document-wide unique identifier for this definition.

## 2.19. <displayreference>

This element gives a mapping between the anchor of a reference and a name that will be displayed instead. This allows authors to display more mnemonic anchor names for automatically-included references. For example, if the reference uses the anchor "RFC6949", the following would cause that anchor in the body of displayed documents to be "RFC-dev":

```
<displayreference target="RFC6449" to="RFC-dev"/>
```

If a reference section is sorted, this element changes the sort order.

This element appears as a child element of: <back> (Section 2.8).

Content model: this element does not have any contents.

### 2.19.1. 'target' attribute (mandatory)

This attribute must be the name of an anchor in a <reference> element.

### 2.19.2. 'to' attribute (mandatory)

This attribute is a name that will be displayed as the anchor instead of the anchor that is given in the <reference> element. The string given must start with one of the following characters: 0-9, a-z, A-Z. The other characters in the string must be 0-9, a-z, A-Z, "-", ".", and "\_".

## 2.20. <dl>

A definition list. Each entry has a pair of elements: a term (<dt>) and a definition (<dd>). (This is slightly different than the model used in HTML, which allows for multiple terms for a single definition.)

This element appears as a child element of: <abstract> (Section 2.1), <aside> (Section 2.6), <blockquote> (Section 2.10), <dd> (Section 2.18), <li> (Section 2.29), <note> (Section 2.33), <section> (Section 2.46), <td> (Section 2.56), and <th> (Section 2.58).

Content model:

One or more sequences of:

1. One <dt> element

2. One <dd> element

#### 2.20.1. 'anchor' attribute

Document-wide unique identifier for the list.

#### 2.20.2. 'hanging' attribute

The hanging attribute defines whether or not the term appears on the same line as the definition. hanging="true" indicates that the term is to the left of the definition, while hanging="false" indicates that the term will be on a separate line.

Allowed values:

- o "false"
- o "true" (default)

#### 2.20.3. 'spacing' attribute

Defines whether or not there is a blank line between entries. spacing="normal" indicates a single blank line, while spacing="compact" indicates no space between.

Allowed values:

- o "normal" (default)
- o "compact"

#### 2.21. <dt>

The term being defined in a definition list.

This element appears as a child element of: <dl> (Section 2.20).

Content model:

In any order:

- o Text
- o <bcp14> elements (Section 2.9)

- o `<cref>` elements (Section 2.16)
- o `<em>` elements (Section 2.22)
- o `<eref>` elements (Section 2.24)
- o `<iref>` elements (Section 2.27)
- o `<relref>` elements (Section 2.44)
- o `<strong>` elements (Section 2.50)
- o `<sub>` elements (Section 2.51)
- o `<sup>` elements (Section 2.52)
- o `<tt>` elements (Section 2.62)
- o `<xref>` elements (Section 2.66)

#### 2.21.1. 'anchor' attribute

Document-wide unique identifier for this term.

#### 2.22. `<em>`

Indicates text that is semantically emphasized. This element will be displayed as italic after processing. This element can be combined with other character formatting elements, and the formatting will be additive.

This element appears as a child element of: `<annotation>` (Section 2.3), `<blockquote>` (Section 2.10), `<cref>` (Section 2.16), `<dd>` (Section 2.18), `<dt>` (Section 2.21), `<li>` (Section 2.29), `<preamble>` (Section 3.6), `<refcontent>` (Section 2.39), `<strong>` (Section 2.50), `<sub>` (Section 2.51), `<sup>` (Section 2.52), `<t>` (Section 2.53), `<td>` (Section 2.56), `<th>` (Section 2.58), and `<tt>` (Section 2.62).

Content model:

In any order:

- o Text
- o `<bcp14>` elements (Section 2.9)

- o <cref> elements (Section 2.16)
- o <eref> elements (Section 2.24)
- o <iref> elements (Section 2.27)
- o <relref> elements (Section 2.44)
- o <strong> elements (Section 2.50)
- o <sub> elements (Section 2.51)
- o <sup> elements (Section 2.52)
- o <tt> elements (Section 2.62)
- o <xref> elements (Section 2.66)

### 2.23. <email>

Provides an email address.

The value is expected to be the addr-spec defined in Section 2 of [RFC6068].

This element appears as a child element of: <address> (Section 2.2).

Content model: only text content.

#### 2.23.1. 'ascii' attribute

The ASCII equivalent of the author's email address. This is only used if the email address has one or two internationalized components.

### 2.24. <eref>

Represents an "external" link (as specified in the "target" attribute). This is useful for embedding URIs in the body of a document.

If the <eref> element has non-empty text content, formatters should use the content as the displayed text that is linked. Otherwise the formatter should use the value of the "target" attribute as the displayed text. Formatters will link the displayed text to the value of the "target" attribute in a manner appropriate for the output format.

For example, with an input of:

```
This is described at
<eref target="http://www.example.com/reports/r12.html"/>.
```

An HTML formatter might generate

```
This is described at
<a href="http://www.example.com/reports/r12.html">
http://www.example.com/reports/r12.html</a>.
```

With an input of:

```
This is described
<eref target="http://www.example.com/reports/r12.html">
in this interesting report</eref>.
```

An HTML formatter might generate

```
This is described
<a href="http://www.example.com/reports/r12.html">
in this interesting report</a>.
```

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <c> (Section 3.1), <cref> (Section 2.16), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <name> (Section 2.32), <postamble> (Section 3.5), <preamble> (Section 3.6), <strong> (Section 2.50), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), <tt> (Section 2.62), and <ttdcol> (Section 3.9).

Content model: only text content.

#### 2.24.1. 'target' attribute (mandatory)

URI of the link target (see Section 3 of [RFC3986]). This must begin with a scheme name (such as "https://") and thus not be relative to the URL of the current document.

#### 2.25. <figure>

Contains a figure with a caption with the figure number. If the element contains a <name> element, the caption will also show that name.

This element appears as a child element of: <aside> (Section 2.6), <blockquote> (Section 2.10), <dd> (Section 2.18), <li>

(Section 2.29), <section> (Section 2.46), <td> (Section 2.56), and <th> (Section 2.58).

Content model:

In this order:

1. One optional <name> element (Section 2.32)
2. Optional <iref> elements (Section 2.27)
3. One optional <preamble> element (Section 3.6)
4. In any order, but at least one of:
  - \* <artwork> elements (Section 2.5)
  - \* <sourcecode> elements (Section 2.48)
5. One optional <postamble> element (Section 3.5)

#### 2.25.1. 'align' attribute

Deprecated.

Note: does not affect title or <artwork> alignment.

Allowed values:

- o "left" (default)
- o "center"
- o "right"

#### 2.25.2. 'alt' attribute

Deprecated. If the goal is to provide a single URI for a reference, use the "target" attribute on <reference> instead.

#### 2.25.3. 'anchor' attribute

Document-wide unique identifier for this figure.

## 2.25.4. 'height' attribute

Deprecated.

## 2.25.5. 'src' attribute

Deprecated.

## 2.25.6. 'suppress-title' attribute

Deprecated. Figures always now get captions.

Allowed values:

- o "true"
- o "false" (default)

## 2.25.7. 'title' attribute

Deprecated. Use <name> instead.

## 2.25.8. 'width' attribute

Deprecated.

## 2.26. &lt;front&gt;

Represent the "front matter": metadata (such as author information), abstract, and additional notes.

A <front> element may have more than one <seriesInfo> elements. A <seriesInfo> element determines the document number (for RFCs) or name (for Internet-Drafts). Another <seriesInfo> element determines the "maturity level" (see Section 4 of [RFC2026]), using values of "std" for "Standards Track", "bcp" for "BCP", "info" for "Informational", "exp" for "Experimental", and "historic" for "Historic". The "name" attributes of those multiple <seriesInfo> elements interact as described in the section on <seriesInfo>.

This element appears as a child element of: <reference> (Section 2.40), and <rfc> (Section 2.45).

Content model:

In this order:



1. One <title> element (Section 2.60)
2. One or more <author> elements (Section 2.7)
3. One optional <date> element (Section 2.17)
4. Optional <area> elements (Section 2.4)
5. Optional <workgroup> elements (Section 2.65)
6. Optional <keyword> elements (Section 2.28)
7. One optional <abstract> element (Section 2.1)
8. Optional <seriesInfo> elements (Section 2.47)
9. Optional <note> elements (Section 2.33)
10. One optional <boilerplate> element (Section 2.11)

## 2.27. <iref>

Provides terms for the document's index.

Index entries can be either be regular entries (when just the "item" attribute is given) or nested entries (by specifying "subitem" as well), grouped under a regular entry.

Index entries generally refer to the exact place where the <iref> element occurred. An exception is the occurrence as a child element of <section>, in which case the whole section is considered to be relevant for that index entry. In some formats, index entries of this type might be displayed as range.

When the prep tool is creating index content, it collects the items in a case-sensitive fashion for both the item and subitem level.

This element appears as a child element of: <annotation> (Section 2.3), <aside> (Section 2.6), <blockquote> (Section 2.10), <c> (Section 3.1), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <figure> (Section 2.25), <li> (Section 2.29), <postamble> (Section 3.5), <preamble> (Section 3.6), <section> (Section 2.46), <strong> (Section 2.50), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <table> (Section 2.54), <td> (Section 2.56), <th> (Section 2.58), <tt> (Section 2.62), and <ttcol> (Section 3.9).

Content model: this element does not have any contents.

#### 2.27.1. 'item' attribute (mandatory)

The item to include.

#### 2.27.2. 'primary' attribute

Setting this to "true" declares the occurrence as "primary", which might cause it to be highlighted in the index.

Allowed values:

- o "true"
- o "false" (default)

#### 2.27.3. 'subitem' attribute

The subitem to include.

#### 2.28. <keyword>

Specifies a keyword applicable to the document.

Note that each element should only contain a single keyword; for multiple keywords, the element can simply be repeated.

Keywords are used both in the RFC Index and in the metadata of generated document representations.

This element appears as a child element of: <front> (Section 2.26).

Content model: only text content.

#### 2.29. <li>

A list element, used in <ol> and <ul>.

This element appears as a child element of: <ol> (Section 2.34), and <ul> (Section 2.63).

Content model:

Either:

In any order, but at least one of:

- \* <artwork> elements (Section 2.5)
- \* <dl> elements (Section 2.20)
- \* <figure> elements (Section 2.25)
- \* <ol> elements (Section 2.34)
- \* <sourcecode> elements (Section 2.48)
- \* <t> elements (Section 2.53)
- \* <ul> elements (Section 2.63)

Or:

In any order, but at least one of:

- \* Text
- \* <bcp14> elements (Section 2.9)
- \* <cref> elements (Section 2.16)
- \* <em> elements (Section 2.22)
- \* <eref> elements (Section 2.24)
- \* <iref> elements (Section 2.27)
- \* <relref> elements (Section 2.44)
- \* <strong> elements (Section 2.50)
- \* <sub> elements (Section 2.51)
- \* <sup> elements (Section 2.52)
- \* <tt> elements (Section 2.62)
- \* <xref> elements (Section 2.66)

### 2.29.1. 'anchor' attribute

Document-wide unique identifier for this list item.

### 2.30. <link>

A link to an external document that is related to the RFC.

The following are the supported types of external documents that can be pointed to in a <link> element:

- o The current ISSN for the RFC Series. The value for the "rel" attribute is "item". The link should use the form "urn:issn:".
- o The DOI for this document. The value for the "rel" attribute is "describedBy". The link should use the form specified in [DOI].
- o The Internet Draft that was submitted to the RFC Editor to become the published RFC. The value for the "rel" attribute is "derivedFrom". The link should be to an IETF-controlled web site that retains copies of Internet Drafts.
- o A representation of the document offered by the document author. The value for the "rel" attribute is "alternate". The link can be to a personally-run web site.

In RFC production mode, the prep tool needs to check the values for <link> before an RFC is published. In draft production mode, the prep tool might remove some <link> elements during the draft submission process.

This element appears as a child element of: <rfc> (Section 2.45).

Content model: this element does not have any contents.

#### 2.30.1. 'href' attribute (mandatory)

The URI of the external document.

#### 2.30.2. 'rel' attribute

The relationship of the external document to this one. The relationships are taken from Link Relations registry maintained by IANA [LINKRELATIONS].

### 2.31. <middle>

Represents the main content of the document.

This element appears as a child element of: <rfc> (Section 2.45).

Content model:

One or more <section> elements (Section 2.46)

### 2.32. <name>

The name of the section, note, figure, or texttable. This name can have flow markup such as to make some characters use a fixed-width font, or to include references.

This element appears as a child element of: <figure> (Section 2.25), <note> (Section 2.33), <references> (Section 2.42), <section> (Section 2.46), <table> (Section 2.54), and <texttable> (Section 3.8).

Content model:

In any order:

- o Text
- o <cref> elements (Section 2.16)
- o <eref> elements (Section 2.24)
- o <relref> elements (Section 2.44)
- o <tt> elements (Section 2.62)
- o <xref> elements (Section 2.66)

### 2.33. <note>

Creates an unnumbered section that appears after the abstract.

It is usually used for additional information to reviewers (working group information, mailing list, ...), or for additional publication information such as "IESG Notes".

This element appears as a child element of: <front> (Section 2.26).

Content model:

In this order:

1. One optional <name> element (Section 2.32)
2. In any order, but at least one of:

- \* <dl> elements (Section 2.20)
- \* <ol> elements (Section 2.34)
- \* <t> elements (Section 2.53)
- \* <ul> elements (Section 2.63)

#### 2.33.1. 'removeInRFC' attribute

If set to "true", this note is marked in the prep tool with text indicating that it should be removed before the document is published as an RFC.

Allowed values:

- o "true"
- o "false" (default)

#### 2.33.2. 'title' attribute

Deprecated. Use <name> instead.

#### 2.34. <ol>

An ordered list. The labels on the items will be either a number or a letter, depending on the value of the style attribute.

This element appears as a child element of: <abstract> (Section 2.1), <aside> (Section 2.6), <blockquote> (Section 2.10), <dd> (Section 2.18), <li> (Section 2.29), <note> (Section 2.33), <section> (Section 2.46), <td> (Section 2.56), and <th> (Section 2.58).

Content model:

One or more <li> elements (Section 2.29)

#### 2.34.1. 'anchor' attribute

Document-wide unique identifier for the list.

#### 2.34.2. 'group' attribute

When the prep tool sees an <ol> element with a "group" attribute that has already been seen, it continues the numbering of the list from where the previous list with the same group name left off. If an <ol> element has both a "group" and "start" attribute, the group's numbering is reset to the given start value.

#### 2.34.3. 'spacing' attribute

Defines whether or not there is a blank line between entries. spacing="normal" indicates a single blank line, while spacing="compact" indicates no space between.

Allowed values:

- o "normal" (default)
- o "compact"

#### 2.34.4. 'start' attribute

The ordinal value to start the list at. This defaults to "1", and must be an integer of 0 or greater.

#### 2.34.5. 'type' attribute

The type of the labels on list items. If the length of the type value is 1, the meaning is the same as it is for HTML:

- a Lowercase letters (a, b, c, ...)
- A Uppercase letters (A, B, C, ...)
- 1 Decimal numbers (1, 2, 3, ...)
- i Lowercase Roman numerals (i, ii, iii, ...)
- I Uppercase Roman numerals (I, II, III, ...)

For type "a" and "A", after the 26th entry, the numbering starts at "aa"/"AA", then "ab"/"AB", and so on.

If the length of the type value is greater than 1, the value must

contain a percent-encoded indicator and other text. The value is a free-form text that allows counter values to be inserted using a "percent-letter" format. For instance, "[REQ%d]" generates labels of the form "[REQ1]", where "%d" inserts the item number as decimal number.

The following formats are supported:

%c Lowercase letters (a, b, c, ...)

%C Uppercase letters (A, B, C, ...)

%d Decimal numbers (1, 2, 3, ...)

%i Lowercase Roman numerals (i, ii, iii, ...)

%I Uppercase Roman numerals (I, II, III, ...)

%% Represents a percent sign

Other formats are reserved for future use. Only one percent encoding other than "%" is allowed in a type string.

It is an error for the type string to be empty. For bulleted lists, you use the <ul> element. For lists that have neither bullets nor numbers, use the <ul> element with the 'empty="true"' attribute.

If no type attribute is given, the default type is the same as "type='%d.'".

## 2.35. <organization>

Specifies the affiliation of an author.

This information appears in both the "Author's Address" section and on the front page (see [RFC7322] for more information). If the value is long, an abbreviated variant can be specified in the "abbrev" attribute.

This element appears as a child element of: <author> (Section 2.7).

Content model: only text content.

### 2.35.1. 'abbrev' attribute

Abbreviated variant.



## 2.35.2. 'ascii' attribute

The ASCII equivalent of the organization's name.

## 2.36. &lt;phone&gt;

Represents a phone number.

The value is expected to be the scheme-specific part of a "tel" URI (so does not include the prefix "tel:"), using the "global numbers" syntax. See Section 3 of [RFC3966] for details.

This element appears as a child element of: <address> (Section 2.2).

Content model: only text content.

## 2.37. &lt;postal&gt;

Contains optional child elements providing postal information. These elements will be displayed in an order that is specific to formatters. A postal address can contain only a set of <street>, <city>, <region>, <code>, and <country> elements, or only an ordered set of <postalLine> elements, but not both.

This element appears as a child element of: <address> (Section 2.2).

Content model:

Either:

In any order:

- \* <city> elements (Section 2.13)
- \* <code> elements (Section 2.14)
- \* <country> elements (Section 2.15)
- \* <region> elements (Section 2.43)
- \* <street> elements (Section 2.49)

Or:

One or more `<postalLine>` elements (Section 2.38)

#### 2.38. `<postalLine>`

Represents one line of a postal address. When more than one `<postalLine>` is given, the prep tool emits them in the order given.

This element appears as a child element of: `<postal>` (Section 2.37).

Content model: only text content.

##### 2.38.1. 'ascii' attribute

The ASCII equivalent of the text in the address line.

#### 2.39. `<refcontent>`

Text that should appear between the title and the date of a reference. The purpose of this element is to prevent the need to abuse `<seriesInfo>` to get such text in a reference.

For example:

```
<reference anchor="Aprill">
  <front>
    <title>On Being A Fool</title>
    <author initials="K." surname="Phunny" fullname="Knot Phunny"/>
    <date year="2000" month="April"/>
  </front>
  <refcontent>Self-published pamphlet</refcontent>
</reference>
```

would render as:

```
[Aprill]      Phunny, K., "On Being A Fool", Self-published
               pamphlet, April 2000.
```

This element appears as a child element of: `<reference>` (Section 2.40).

Content model:

In any order:

- o Text
- o `<bcpl4>` elements (Section 2.9)

- o `<em>` elements (Section 2.22)
- o `<strong>` elements (Section 2.50)
- o `<sub>` elements (Section 2.51)
- o `<sup>` elements (Section 2.52)
- o `<tt>` elements (Section 2.62)

#### 2.40. `<reference>`

Represents a bibliographical reference.

This element appears as a child element of: `<referencegroup>` (Section 2.41), and `<references>` (Section 2.42).

Content model:

In this order:

1. One `<front>` element (Section 2.26)
2. In any order:
  - \* `<annotation>` elements (Section 2.3)
  - \* `<format>` elements (Section 3.3)
  - \* `<refcontent>` elements (Section 2.39)
  - \* `<seriesInfo>` elements (Section 2.47; deprecated in this context)

##### 2.40.1. 'anchor' attribute (mandatory)

Document-wide unique identifier for this reference. Usually, this will be used both to "label" the reference in the references section, and as an identifier in links to this reference entry.

##### 2.40.2. 'quoteTitle' attribute

Specifies whether or not the title in the reference should be quoted. This can be used to prevent quoting, such as on errata.

Allowed values:

- o "true" (default)
- o "false"

#### 2.40.3. 'target' attribute

Holds the URI for the reference.

#### 2.41. <referencegroup>

Represents a list of bibliographic references that will be represented as a single reference. This is most often used for references in the STD and BCP series, where a single reference (such as "BCP 9") encompasses more than one RFC.

This element appears as a child element of: <references> (Section 2.42).

Content model:

One or more <reference> elements (Section 2.40)

##### 2.41.1. 'anchor' attribute (mandatory)

Document-wide unique identifier for this reference group. Usually, this will be used both to "label" the reference group in the references section, and as an identifier in links to this reference entry.

#### 2.42. <references>

Contains a set of bibliographical references.

In the early days of the RFC series, there was only one "References" section per RFC. This convention was later changed to group references into two sets, "Normative" and "Informative" as described in [RFC7322]). This vocabulary supports the split with the <name> child element. In general, the title should be either "Normative References" or "Informative References".

By default, the order of references is significant. Some formatters, however, might be able to be requested to sort them based on their anchor names.

This element appears as a child element of: <back> (Section 2.8).

Content model:

In this order:

1. One optional `<name>` element (Section 2.32)
2. In any order:
  - \* `<reference>` elements (Section 2.40)
  - \* `<referencegroup>` elements (Section 2.41)

#### 2.42.1. 'anchor' attribute

An optional user-supplied for this section.

#### 2.42.2. 'title' attribute

Deprecated. Use `<name>` instead.

#### 2.43. `<region>`

Provides the region name in a postal address.

This element appears as a child element of: `<postal>` (Section 2.37).

Content model: only text content.

##### 2.43.1. 'ascii' attribute

The ASCII equivalent of the region name.

#### 2.44. `<relref>`

A relative link to a reference from the References section. Formatters that have links (such as HTML and PDF) are likely to render `<relref>` elements as external hyperlinks to the specified part of the reference, creating the link target by combining the base URI from the `<reference>` element with the "relative" attribute from this element. The "target" attribute is required, and it must be the anchor of a `<reference>` element.

Either the "relative" or the "section" attribute must be present, but both cannot be given for a `<relref>` element. If a reference is an RFC or Internet-Draft that is in the v3 format and the desired relative reference is to a section of that reference, the "section" attribute is easier to use than the "relative" attribute because the value of "section" is just a section string such as "2.3".

An example of the <relref> element with text content might be:

```
See <relref section="2.3" target="RFC7878">
the protocol overview</relref>
for more information.
```

An HTML formatter might generate:

```
See
<a href="http://www.rfc-editor.org/info/rfc7878#s-2.3">
the protocol overview</a>
for more information.
```

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <cref> (Section 2.16), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <name> (Section 2.32), <preamble> (Section 3.6), <strong> (Section 2.50), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), and <tt> (Section 2.62).

Content model: only text content.

#### 2.44.1. 'displayFormat' attribute

This attribute is used to signal formatters what the desired format of the relative reference should be. Formatters for document types that have linking capability should wrap each part of the displayed text in hyperlinks. If there is content in the <relref> element, formatters will ignore the value of this attribute.

"of"

A formatter should display the relative reference as the contents of the "derivedRemoteContent" attribute followed by a space, the word "of", another space, and the value from the "target" attribute enclosed in square brackets.

For example, if Section 2.3 of RFC 7878 has the title "Protocol Overview", with an input of:

```
See
<relref section="2.3" target="RFC7878" displayFormat="of"/>
for an overview.
```

An HTML formatter might generate:

See  
<a href="http://www.rfc-editor.org/info/rfc7878#s-2.3">  
Section 2.3</a> of  
[<a href="#RFC7878">RFC7878</a>]  
for an overview.

Note that "displayFormat='of'" is the default for <relref> so it does not need to be given in a <relref> element if that format is desired.

"comma"

A formatter should display the relative reference as the value from the "target" attribute enclosed in square brackets, a comma, a space, and the "derivedRemoteContent" attribute.

For example, if Section 2.3 of RFC 7878 has the title "Protocol Overview", with an input of:

See  
<relref section="2.3" target="RFC7878" displayFormat="comma"/>,  
for an overview.

An HTML formatter might generate:

See  
[<a href="#RFC7878">RFC7878</a>],  
<a href="http://www.rfc-editor.org/info/rfc7878#s-2.3">  
Section 2.3</a>, for an overview.

"parens"

A formatter should display the relative reference as the value from the "target" attribute enclosed in square brackets, a space, a left parenthesis, the "derivedRemoteContent" attribute, and a right parenthesis.

For example, if Section 2.3 of RFC 7878 has the title "Protocol Overview", with an input of:

See  
<relref section="2.3" target="RFC7878" displayFormat="parens"/>  
for an overview.

An HTML formatter might generate

See  
[<a href="#RFC7878">RFC7878</a>]  
(<a href="http://www.rfc-editor.org/info/rfc7878#s-2.3">  
Section 2.3</a>)  
for an overview.

"bare"

A formatter should display the relative reference as the contents of the "derivedRemoteContent" attribute and nothing else. This is useful when there are multiple relative references to a single base reference.

For example:

See  
<relref section="2.3" target="RFC7878" displayFormat="bare"/>  
and  
<relref section="2.4" target="RFC7878" displayFormat="of"/>  
for an overview.

An HTML formatter might generate:

See  
<a href="http://www.rfc-editor.org/info/rfc7878#s-2.3">  
Section 2.3</a>  
and  
<a href="http://www.rfc-editor.org/info/rfc7878#s-2.4">  
Section 2.4</a> of  
[<a href="#RFC7878">RFC7878</a>]  
for an overview.

Allowed values:

- o "of" (default)
- o "comma"
- o "parens"
- o "bare"

#### 2.44.2. 'relative' attribute

Specifies a relative reference from the URI in the target reference. This value must include whatever leading character is needed to create the relative reference; typically, this is "#" for HTML documents.



#### 2.44.3. 'section' attribute

Specifies a section of the target reference. If the reference is not an RFC or Internet-Draft, it is an error.

#### 2.44.4. 'target' attribute (mandatory)

The anchor of the reference for this element. If this value is not an anchor to a <reference> or <referencegroup> element, it is an error. If the reference at the target has no URI, it is an error.

#### 2.45. <rfc>

This is the root element of the xml2rfc vocabulary.

Content model:

In this order:

1. Optional <link> elements (Section 2.30)
2. One <front> element (Section 2.26)
3. One <middle> element (Section 2.31)
4. One optional <back> element (Section 2.8)

##### 2.45.1. 'category' attribute

Deprecated; instead, use the "name" attribute in <seriesInfo>.

##### 2.45.2. 'consensus' attribute

Affects the generated boilerplate. Note that the values of "no" and "yes" are deprecated and are replaced by "false" (the default) and "true".

See [RFC5741] for more information.

Allowed values:

- o "no"
- o "yes"
- o "false" (default)

- o "true"

#### 2.45.3. 'docName' attribute

Deprecated; instead, use the "value" attribute in <seriesInfo>.

#### 2.45.4. 'indexInclude' attribute

Specifies whether or not a formatter is requested to include an index in generated files. If the source file has no <iref> elements, an index is never generated. This option is useful for generating documents where the source document has <iref> elements but the author no longer wants an index.

Allowed values:

- o "true" (default)
- o "false"

#### 2.45.5. 'ipr' attribute

Represents the Intellectual Property status of the document. See Appendix A.1 for details.

Allowed values:

- o "full2026"
- o "noDerivativeWorks2026"
- o "none"
- o "full3667"
- o "noModification3667"
- o "noDerivatives3667"
- o "full3978"
- o "noModification3978"
- o "noDerivatives3978"
- o "trust200811"

- o "noModificationTrust200811"
- o "noDerivativesTrust200811"
- o "trust200902"
- o "noModificationTrust200902"
- o "noDerivativesTrust200902"
- o "pre5378Trust200902"

#### 2.45.6. 'iprExtract' attribute

Identifies a single section within the document for which extraction "as-is" is explicitly allowed (only relevant for historic values of the "ipr" attribute).

#### 2.45.7. 'number' attribute

Deprecated; instead, use the "value" attribute in <seriesInfo>.

#### 2.45.8. 'obsoletes' attribute

A comma-separated list of RFC numbers or Internet-Draft names.

The prep tool will parse the attribute value so that incorrect references can be detected.

#### 2.45.9. 'prepTime' attribute

The date that the XML was processed by a preparation tool. This is included in the XML file just before it is saved to disk. The value is formatted using the format from [RFC3339].

#### 2.45.10. 'seriesNo' attribute

Deprecated; instead, use the "value" attribute in <seriesInfo>.

#### 2.45.11. 'sortRefs' attribute

Specifies whether or not the prep tool will sort the references in each reference section.

Allowed values:

- o "true"
- o "false" (default)

#### 2.45.12. 'submissionType' attribute

The document stream.

See Section 2 of [RFC5741] for details.

Allowed values:

- o "IETF" (default)
- o "IAB"
- o "IRTF"
- o "independent"

#### 2.45.13. 'symRefs' attribute

Specifies whether or not a formatter is requested to use symbolic references (such as "[RFC2119]"). If the value for this is "false", the references come out as numbers (such as "[3]").

Allowed values:

- o "true" (default)
- o "false"

#### 2.45.14. 'tocDepth' attribute

Specifies number of levels of heading that formatter is requested to include in the table of contents; the default is "3".

#### 2.45.15. 'tocInclude' attribute

Specifies whether or not a formatter is requested to include a table of contents in generated files.

Allowed values:

- o "true" (default)
- o "false"

#### 2.45.16. 'updates' attribute

A comma-separated list of RFC numbers or Internet-Draft names.

The prep tool will parse the attribute value so that incorrect references can be detected.

#### 2.45.17. 'version' attribute

Specifies the version of xml2rfc syntax used in this document. The only expected value (for now) is "3".

#### 2.46. <section>

Represents a section (when inside a <middle> element) or an appendix (when inside a <back> element).

Sub-sections are created by nesting <section> elements inside <section> elements. Sections are allowed to be empty.

This element appears as a child element of: <back> (Section 2.8), <boilerplate> (Section 2.11), <middle> (Section 2.31), and <section> (Section 2.46).

Content model:

In this order:

1. One optional <name> element (Section 2.32)
2. In any order:
  - \* <artwork> elements (Section 2.5)
  - \* <aside> elements (Section 2.6)
  - \* <blockquote> elements (Section 2.10)
  - \* <dl> elements (Section 2.20)
  - \* <figure> elements (Section 2.25)
  - \* <iref> elements (Section 2.27)
  - \* <ol> elements (Section 2.34)

- \* `<sourcecode>` elements (Section 2.48)
- \* `<t>` elements (Section 2.53)
- \* `<table>` elements (Section 2.54)
- \* `<texttable>` elements (Section 3.8)
- \* `<ul>` elements (Section 2.63)

### 3. Optional `<section>` elements (Section 2.46)

#### 2.46.1. 'anchor' attribute

Document-wide unique identifier for this section.

#### 2.46.2. 'numbered' attribute

If set to "false", the formatter is requested to not display a section number. The prep tool will verify that such a section is not followed by a numbered section in a part, and will verify that the section is a top-level section.

Allowed values:

- o "true" (default)
- o "false"

#### 2.46.3. 'removeInRFC' attribute

If set to "true", the formatter is requested to mark this section with a paragraph at the beginning of the section indicating that it should be removed before the document is published as an RFC.

Allowed values:

- o "true"
- o "false" (default)

#### 2.46.4. 'title' attribute

Deprecated. Use `<name>` instead.

#### 2.46.5. 'toc' attribute

Indicates to a formatter whether or not the section is to be included in a table of contents, if such a table of contents is produced. This only takes effect if the level of the section would have appeared in the table of contents based on the "tocDepth" attribute of the <rfc> element, and of course only if the table of contents is being created based on the "tocInclude" attribute of the <rfc> element. If this is set to "exclude", any section below this one will be excluded as well. The "default" value indicates to include the section if it would be included by the tocDepth attribute of the <rfc> element.

Allowed values:

- o "include"
- o "exclude"
- o "default" (default)

#### 2.47. <seriesInfo>

Specifies the document series in which this document appears, and also specifies an identifier within that series.

A processing tool determines if it is working on an RFC or an Internet-Draft by inspecting the "name" attribute of a <seriesInfo> element inside the <front> element inside the <rfc> element, looking for "rfc" or "Internet-Draft". (Specifying neither value in any of the <seriesInfo> elements can be useful for producing other types of documents, but is out-of-scope for this specification.)

It is invalid to have multiple <seriesInfo> elements inside the <front> element inside the <rfc> element that have "name" values of "rfc", or that have "name" values of "Internet-Draft", or that have both "rfc" and "Internet-Draft".

This element appears as a child element of: <front> (Section 2.26), and <reference> (Section 2.40; deprecated in this context).

Content model: this element does not have any contents.

##### 2.47.1. 'asciiName' attribute

The ASCII equivalent of the name field.

#### 2.47.2. 'asciiValue' attribute

The ASCII equivalent of the value field.

#### 2.47.3. 'name' attribute (mandatory)

The name of the series.

Some of the values for "name" interact as follows:

- o If a <front> element contains a <seriesInfo> element with a name of "Internet-Draft", it can also have at most one additional <seriesInfo> element with a name of "std", "bcp", "fyi", "exp", or "historic" to indicate the intended status of this Internet Draft, if it were to be later published as an RFC. If such an additional <seriesInfo> element has one of those names, the value attribute for that name MUST be "".
- o If a <front> element contains a <seriesInfo> element with a name of "rfc", it can also have at most one additional <seriesInfo> element with a name of "std", "bcp", or "fyi" to indicate the current status of this RFC. If such an additional <seriesInfo> element has one of those names, the value attribute for that name MUST be the number within that series. That <front> element might also contain an additional <seriesInfo> with the name "exp" or "historic" and a value of "" to indicate the status of the RFC.
- o A <front> element that has a <seriesInfo> element that has the name "Internet-Draft" MUST NOT also have a <seriesInfo> element that has the name "rfc".
- o The DOI for the referenced document. This MUST NOT be used when <seriesInfo> element is an eventual child element of a <rfc> element, only as an eventual child of a <reference> element. The value attribute should use the form specified in [DOI].

Other values might be added at a later time by the RFC Editor.

#### 2.47.4. 'value' attribute (mandatory)

The identifier within the series specified by the "name" attribute.

For BCPs, FYIs, RFCs, and STDs this is the number within the series. For Internet-Drafts, it is the full draft name (ending with the two-digit version number). For DOIs, the value is given such as "10.123456/rfc1149", (the actual value will be specified later in [DOI]).



The name in the value should be the document name without any file extension. For Internet Drafts, the value for this attribute should be "draft-ietf-somewg-someprotocol-07", not "draft-ietf-somewg-someprotocol-07.txt".

#### 2.48. <sourcecode>

This element allows the inclusion of sourcecode into the document.

<sourcecode> provides full control of horizontal whitespace and line breaks. It is thus useful for source code and formal languages (such as ABNF or the RNC notation used in this document). When rendered, sourcecode is always shown in a monospace font.

For artwork such as character-based art, diagrams of message layouts, and so on, use the <artwork> element instead.

A common problem authors have with <sourcecode> is that the XML processor returns errors if the text in the artwork contains either the "&" or "<" character, or the string "]]>". To avoid these problems, the "&" and "<" characters may be escaped using the strings "&amp;" and "&lt;", respectively; the "]]>" string can be represented as "]]&gt;". Alternatively, they may be surrounded in a CDATA structure: "<![CDATA[ ]]>". For example:

Desired output:

```
allowed-chars = "." | "," | "&" | "<" | ">" | "|" | "
```

Using escaping:

```
<sourcecode>
  allowed-chars = "." | "," | "&amp;" | "&lt;" | "&gt;" | "|" | "
</sourcecode>
```

Using CDATA:

```
<sourcecode>
<![CDATA[  allowed-chars = "." | "," | "&" | "<" | ">" | "|" ]]>
</sourcecode>
```

Using CDATA is not a panacea, but it does help prevent having to use escapes in places where using escapes can cause other problems, such as difficulty of inclusion from other documents.

Output formatters that do pagination should attempt to keep source code on a single page. This is to prevent source code that is split across pages from looking like two separate pieces of code.

This element appears as a child element of: <blockquote> (Section 2.10), <dd> (Section 2.18), <figure> (Section 2.25), <li>

(Section 2.29), <section> (Section 2.46), <td> (Section 2.56), and <th> (Section 2.58).

Content model: only text content.

#### 2.48.1. 'anchor' attribute

Document-wide unique identifier for this sourcecode.

#### 2.48.2. 'name' attribute

A filename suitable for the contents (such as for extraction to a local file). This attribute can be helpful for other kinds of tools (such as automated syntax checkers which work by extracting the source code). Note that the "name" attribute does not need to be unique for artwork elements in a document. If multiple sourcecode elements have the same name attribute, a formatter might assume that the elements are all fragments of a single file, and such a formatter can collect those fragments for later processing.

#### 2.48.3. 'src' attribute

The URI reference of a source file ([RFC3986]).

It is an error to have both a "src" attribute and content in the <sourcecode> element.

#### 2.48.4. 'type' attribute

Specifies the type of the sourcecode. The value of this attribute is free text with certain values designated as preferred.

The preferred values for <sourcecode> types are:

- o abnf
- o asn.1
- o bash
- o c++
- o c
- o cbor
- o dtd

- o java
- o javascript
- o json
- o mib
- o perl
- o pseudocode
- o python
- o rnc
- o xml

The RFC Editor will maintain a complete list of the preferred values on its web site, and that list is expected to be updated over time. Thus, a consumer of v3 XML should not cause a failure when it encounters an unexpected type.

#### 2.49. <street>

Provides a street address.

This element appears as a child element of: <postal> (Section 2.37).

Content model: only text content.

##### 2.49.1. 'ascii' attribute

The ASCII equivalent of the street address.

#### 2.50. <strong>

Indicates text that is semantically strong. This element will be displayed as bold after processing. This element can be combined with other character formatting elements, and the formatting will be additive.

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <cref> (Section 2.16), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <preamble> (Section 3.6), <refcontent> (Section 2.39), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), and <tt>

(Section 2.62).

Content model:

In any order:

- o Text
- o <bcpl4> elements (Section 2.9)
- o <cref> elements (Section 2.16)
- o <em> elements (Section 2.22)
- o <eref> elements (Section 2.24)
- o <iref> elements (Section 2.27)
- o <relref> elements (Section 2.44)
- o <sub> elements (Section 2.51)
- o <sup> elements (Section 2.52)
- o <tt> elements (Section 2.62)
- o <xref> elements (Section 2.66)

#### 2.51. <sub>

Causes the text to be displayed as subscript, approximately half a letter-height lower than normal text. This element can be combined with other character formatting elements, and the formatting will be additive.

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <cref> (Section 2.16), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <preamble> (Section 3.6), <refcontent> (Section 2.39), <strong> (Section 2.50), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), and <tt> (Section 2.62).

Content model:

In any order:

- o Text
- o <bcp14> elements (Section 2.9)
- o <cref> elements (Section 2.16)
- o <em> elements (Section 2.22)
- o <eref> elements (Section 2.24)
- o <iref> elements (Section 2.27)
- o <relref> elements (Section 2.44)
- o <strong> elements (Section 2.50)
- o <tt> elements (Section 2.62)
- o <xref> elements (Section 2.66)

#### 2.52. <sup>

Causes the text to be displayed as superscript, approximately half a letter-height higher than normal text. This element can be combined with other character formatting elements, and the formatting will be additive.

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <cref> (Section 2.16), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <preamble> (Section 3.6), <refcontent> (Section 2.39), <strong> (Section 2.50), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), and <tt> (Section 2.62).

Content model:

In any order:

- o Text
- o <bcp14> elements (Section 2.9)
- o <cref> elements (Section 2.16)
- o <em> elements (Section 2.22)
- o <eref> elements (Section 2.24)

- o `<iref>` elements (Section 2.27)
- o `<relref>` elements (Section 2.44)
- o `<strong>` elements (Section 2.50)
- o `<tt>` elements (Section 2.62)
- o `<xref>` elements (Section 2.66)

#### 2.53. `<t>`

Contains a paragraph of text.

This element appears as a child element of: `<abstract>` (Section 2.1), `<aside>` (Section 2.6), `<blockquote>` (Section 2.10), `<dd>` (Section 2.18), `<li>` (Section 2.29), `<list>` (Section 3.4), `<note>` (Section 2.33), `<section>` (Section 2.46), `<td>` (Section 2.56), and `<th>` (Section 2.58).

Content model:

In any order:

- o Text
- o `<bcpl4>` elements (Section 2.9)
- o `<cref>` elements (Section 2.16)
- o `<em>` elements (Section 2.22)
- o `<eref>` elements (Section 2.24)
- o `<iref>` elements (Section 2.27)
- o `<list>` elements (Section 3.4)
- o `<relref>` elements (Section 2.44)
- o `<spanx>` elements (Section 3.7)
- o `<strong>` elements (Section 2.50)
- o `<sub>` elements (Section 2.51)
- o `<sup>` elements (Section 2.52)

- o `<tt>` elements (Section 2.62)
- o `<vspace>` elements (Section 3.10)
- o `<xref>` elements (Section 2.66)

#### 2.53.1. 'anchor' attribute

Document-wide unique identifier for this paragraph.

#### 2.53.2. 'hangText' attribute

Deprecated. Instead use `<dd>` inside of a definition list (`<dl>`).

#### 2.53.3. 'keepWithNext' attribute

Acts as a hint to the output formatters that do pagination to do a best effort attempt to keep the paragraph with the next element, whatever that happens to be. For example, the HTML output @media print CSS might translate this to page-break-after: avoid. For PDF, the paginator could attempt to keep the paragraph with the next element. Note: this attribute is strictly a hint and not always actionable.

Allowed values:

- o "false" (default)
- o "true"

#### 2.53.4. 'keepWithPrevious' attribute

Acts as a hint to the output formatters that do pagination to do a best effort attempt to keep the paragraph with the previous element, whatever that happens to be. For example, the HTML output @media print CSS might translate this to page-break-before: avoid. For PDF, the paginator could attempt to keep the paragraph with the previous element. Note: this attribute is strictly a hint and not always actionable.

Allowed values:

- o "false" (default)
- o "true"

## 2.54. <table>

Contains a table with a caption with the table number. If the element contains a <name> element, the caption will also show that name.

Inside the <table> element is optionally a <thead> element to contain the rows that will be the table's heading and optionally a <tfoot> element to contain the rows of the table's footer. If the XML is converted to a representation that has page breaks (such as PDFs, or printed HTML), the header and footer are meant to appear on each page.

This element appears as a child element of: <aside> (Section 2.6), and <section> (Section 2.46).

Content model:

In this order:

1. One optional <name> element (Section 2.32)
2. Optional <iref> elements (Section 2.27)
3. One optional <thead> element (Section 2.59)
4. One or more <tbody> elements (Section 2.55)
5. One optional <tfoot> element (Section 2.57)

### 2.54.1. 'anchor' attribute

Document-wide unique identifier for this table.

## 2.55. <tbody>

A container for a set of body rows for a table.

This element appears as a child element of: <table> (Section 2.54).

Content model:

One or more <tr> elements (Section 2.61)

### 2.55.1. 'anchor' attribute

Document-wide unique identifier for the tbody.



## 2.56. &lt;td&gt;

A cell in a table row.

This element appears as a child element of: <tr> (Section 2.61).

Content model:

Either:

One or more <t> elements (Section 2.53)

Or:

In any order, but at least one of:

- \* Text
- \* <artwork> elements (Section 2.5)
- \* <bcp14> elements (Section 2.9)
- \* <br> elements (Section 2.12)
- \* <cref> elements (Section 2.16)
- \* <dl> elements (Section 2.20)
- \* <em> elements (Section 2.22)
- \* <eref> elements (Section 2.24)
- \* <figure> elements (Section 2.25)
- \* <ieref> elements (Section 2.27)
- \* <ol> elements (Section 2.34)
- \* <relref> elements (Section 2.44)
- \* <sourcecode> elements (Section 2.48)
- \* <strong> elements (Section 2.50)
- \* <sub> elements (Section 2.51)

- \* <sup> elements (Section 2.52)
- \* <tt> elements (Section 2.62)
- \* <ul> elements (Section 2.63)
- \* <xref> elements (Section 2.66)

#### 2.56.1. 'align' attribute

Controls whether the content of the cell appears left justified (default), centered, or right justified. Note that "center" or "right" probably only work well in cells with plain text; any other elements might make the contents render badly.

Allowed values:

- o "left" (default)
- o "center"
- o "right"

#### 2.56.2. 'anchor' attribute

Document-wide unique identifier for the cell.

#### 2.56.3. 'border' attribute

The width of the border for this cell. The default is 0, meaning no border.

#### 2.56.4. 'colspan' attribute

The number of columns that the cell to span. For example, setting "colspan='3'" indicates that the cell occupies the same horizontal space as three cells in the row above or below this one.

#### 2.56.5. 'rowspan' attribute

The number of rows that the cell to span. For example, setting "rowspan='3'" indicates that the cell occupies the same vertical space as three rows.

## 2.57. <tfoot>

A container for a set of footer rows for a table.

This element appears as a child element of: <table> (Section 2.54).

Content model:

One or more <tr> elements (Section 2.61)

### 2.57.1. 'anchor' attribute

Document-wide unique identifier for the tfoot.

## 2.58. <th>

A cell in a table row. When rendered, this will normally come out in boldface; other than that, there is no difference between this and the <td> element.

This element appears as a child element of: <tr> (Section 2.61).

Content model:

Either:

One or more <t> elements (Section 2.53)

Or:

In any order, but at least one of:

- \* Text
- \* <artwork> elements (Section 2.5)
- \* <bcp14> elements (Section 2.9)
- \* <br> elements (Section 2.12)
- \* <cref> elements (Section 2.16)
- \* <dl> elements (Section 2.20)
- \* <em> elements (Section 2.22)

- \* <eref> elements (Section 2.24)
- \* <figure> elements (Section 2.25)
- \* <iref> elements (Section 2.27)
- \* <ol> elements (Section 2.34)
- \* <relref> elements (Section 2.44)
- \* <sourcecode> elements (Section 2.48)
- \* <strong> elements (Section 2.50)
- \* <sub> elements (Section 2.51)
- \* <sup> elements (Section 2.52)
- \* <tt> elements (Section 2.62)
- \* <ul> elements (Section 2.63)
- \* <xref> elements (Section 2.66)

#### 2.58.1. 'align' attribute

Controls whether the content of the cell appears left justified (default), centered, or right justified. Note that "center" or "right" probably only work well in cells with plain text; any other elements might make the contents render badly.

Allowed values:

- o "left" (default)
- o "center"
- o "right"

#### 2.58.2. 'anchor' attribute

Document-wide unique identifier for the row.

#### 2.58.3. 'border' attribute

The width of the border for this cell. The default is 0, meaning no border.

#### 2.58.4. 'colspan' attribute

The number of columns that the cell to span. For example, setting "colspan='3'" indicates that the cell occupies the same horizontal space as three cells in the row above or below this one.

#### 2.58.5. 'rowspan' attribute

The number of rows that the cell to span. For example, setting "rowspan='3'" indicates that the cell occupies the same vertical space as three rows.

### 2.59. <thead>

A container for a set of header rows for a table.

This element appears as a child element of: <table> (Section 2.54).

Content model:

One or more <tr> elements (Section 2.61)

#### 2.59.1. 'anchor' attribute

Document-wide unique identifier for the thead.

### 2.60. <title>

Represents the document title.

When this element appears in the <front> element of the current document, the title might also appear in page headers or footers. If it is long (~40 characters), the "abbrev" attribute can be used to specify an abbreviated variant.

This element appears as a child element of: <front> (Section 2.26).

Content model: only text content.

#### 2.60.1. 'abbrev' attribute

Specifies an abbreviated variant of the document title.

#### 2.60.2. 'ascii' attribute

The ASCII equivalent of the title.

## 2.61. <tr>

A row of a table.

This element appears as a child element of: <tbody> (Section 2.55), <tfoot> (Section 2.57), and <thead> (Section 2.59).

Content model:

In any order, but at least one of:

- o <td> elements (Section 2.56)
- o <th> elements (Section 2.58)

### 2.61.1. 'anchor' attribute

Document-wide unique identifier for the row.

### 2.61.2. 'border' attribute

The width of the border for this row. The default is 0, meaning no border. If the cells in this row also have "border" attributes, those values override this value for those cells.

## 2.62. <tt>

Causes the text to be displayed in a constant-width font. This element can be combined with other character formatting elements, and the formatting will be additive.

This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <cref> (Section 2.16), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <name> (Section 2.32), <preamble> (Section 3.6), <refcontent> (Section 2.39), <strong> (Section 2.50), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <td> (Section 2.56), and <th> (Section 2.58).

Content model:

In any order:

- o Text
- o <bcp14> elements (Section 2.9)

- o `<cref>` elements (Section 2.16)
- o `<em>` elements (Section 2.22)
- o `<eref>` elements (Section 2.24)
- o `<iref>` elements (Section 2.27)
- o `<relref>` elements (Section 2.44)
- o `<strong>` elements (Section 2.50)
- o `<sub>` elements (Section 2.51)
- o `<sup>` elements (Section 2.52)
- o `<xref>` elements (Section 2.66)

### 2.63. `<ul>`

An unordered list. The labels on the items will be symbols picked by the formatter.

This element appears as a child element of: `<abstract>` (Section 2.1), `<aside>` (Section 2.6), `<blockquote>` (Section 2.10), `<dd>` (Section 2.18), `<li>` (Section 2.29), `<note>` (Section 2.33), `<section>` (Section 2.46), `<td>` (Section 2.56), and `<th>` (Section 2.58).

Content model:

One or more `<li>` elements (Section 2.29)

#### 2.63.1. 'anchor' attribute

Document-wide unique identifier for the list.

#### 2.63.2. 'empty' attribute

Defines whether or not the label is empty. `empty="true"` indicates that no label be shown.

Allowed values:

- o `"false"` (default)
- o `"true"`

### 2.63.3. 'spacing' attribute

Defines whether or not there is a blank line between entries.  
spacing="normal" indicates a single blank line, while  
spacing="compact" indicates no space between.

Allowed values:

- o "normal" (default)
- o "compact"

### 2.64. <uri>

Contains a web address associated with the author.

The contents should be a valid URI (see Section 3 of [RFC3986]).

This element appears as a child element of: <address> (Section 2.2).

Content model: only text content.

### 2.65. <workgroup>

This element is used to specify the Working Group (IETF) or Research Group (IRTF) from which the document originates, if any. The recommended format is the official name of the Working Group (with some capitalization).

In Internet-Drafts, this is used in the upper left corner of the boilerplate, replacing the "Network Working Group" string. Formatting software can append the words "Working Group" or "Research Group", depending on the "submissionType" property on the <rfc> element (Section 2.45.12).

This element appears as a child element of: <front> (Section 2.26).

Content model: only text content.

### 2.66. <xref>

A reference to an anchor in this document. Formatters that have links (such as HTML and PDF) are likely to render <xref> elements as internal hyperlinks. This element is useful for referring to references in the References section, to specific sections of this document, to specific figures, and so on. The "target" attribute is required.



This element appears as a child element of: <annotation> (Section 2.3), <blockquote> (Section 2.10), <c> (Section 3.1), <cref> (Section 2.16), <dd> (Section 2.18), <dt> (Section 2.21), <em> (Section 2.22), <li> (Section 2.29), <name> (Section 2.32), <postamble> (Section 3.5), <preamble> (Section 3.6), <strong> (Section 2.50), <sub> (Section 2.51), <sup> (Section 2.52), <t> (Section 2.53), <td> (Section 2.56), <th> (Section 2.58), <tt> (Section 2.62), and <ttcol> (Section 3.9).

Content model: only text content.

#### 2.66.1. 'format' attribute

This attribute signals to formatters what the desired format of the reference should be. Formatters for document types that have linking capability should wrap the displayed text in hyperlinks.

"counter"

The "derivedContent" attribute will contain just a counter. This is used for targets that are <section>, <figure>, <table>, or item in an ordered list. Using "format='counter'" where the target is any other type of element is an error.

For example, with an input of:

```
<section anchor="overview">Protocol Overview</section>
. . .
See Section <xref target="overview" format="counter"/>
for an overview.
```

An HTML formatter might generate "See Section <a href="#overview">1.7</a> for an overview."

"default"

The "derivedContent" attribute will contain a text fragment that describes the referenced part completely, such as "XML" for a target that is a <reference>, or "Section 2" or "Table 4" for a target to a non-reference.

For example, with an input of:

```
<section anchor="overview">Protocol Overview</section>
. . .
See <xref target="overview"/> for an overview.
```

An HTML formatter might generate "See <a href="#overview">Section 1.7</a> for an overview."

"none"

Deprecated.

"title"

If the target is a <reference> element, the "derivedContent" attribute will contain the name of the reference, extracted from the <title> child of the <front> child of the reference. Or, if the target element has a <name> child element, the "derivedContent" attribute will contain the text content of that <name> element concatenated with the text content of each descendant node of <name> (that is, stripping out all of the XML markup, leaving only the text). Or, if the target element does not contain a <name> child element, the "derivedContent" attribute will contain the name of the "anchor" attribute of that element with no other adornment.

Allowed values:

- o "default" (default)
- o "title"
- o "counter"
- o "none"

#### 2.66.2. 'pageno' attribute

Deprecated.

Allowed values:

- o "true"
- o "false" (default)

#### 2.66.3. 'target' attribute (mandatory)

Identifies the document component being referenced. The value needs to match the value of the "anchor" attribute of an element in the document, otherwise it is an error.

## 2.67. <svg> (in namespace <http://www.w3.org/2000/svg>)

This element holds SVG art. The use of SVG in Internet Drafts and RFCs is covered in much more detail in [SVGforRFCs].

This element appears as a child element of: <artwork> (Section 2.5).

Content model: this element does not have any contents.

### 2.67.1. 'TheRealDefinition' attribute (mandatory)

(This is just a placeholder until [SVGforRFCs] has complete RNG that can be incorporated by reference in this draft.)

## 3. Elements from v2 That Have Been Deprecated

This section lists the elements from v2 that have been deprecated. Note that some elements in v3 have attributes from v2 that are deprecated; those are not listed here.

### 3.1. <c>

Deprecated. Instead, use <tr>, <td>, and <th>.

This element appears as a child element of: <texttable> (Section 3.8).

Content model:

In any order:

- o Text
- o <bcp14> elements (Section 2.9)
- o <cref> elements (Section 2.16)
- o <em> elements (Section 2.22)
- o <eref> elements (Section 2.24)
- o <iref> elements (Section 2.27)
- o <spanx> elements (Section 3.7)
- o <strong> elements (Section 2.50)

- o `<sub>` elements (Section 2.51)
- o `<sup>` elements (Section 2.52)
- o `<tt>` elements (Section 2.62)
- o `<xref>` elements (Section 2.66)

### 3.2. `<facsimile>`

Deprecated. The `<email>` element is a much more useful way to get in touch with authors.

This element appears as a child element of: `<address>` (Section 2.2).

Content model: only text content.

### 3.3. `<format>`

Deprecated. If the goal is to provide a single URI for a reference, use the "target" attribute on `<reference>` instead.

This element appears as a child element of: `<reference>` (Section 2.40).

Content model: this element does not have any contents.

#### 3.3.1. 'octets' attribute

Deprecated.

#### 3.3.2. 'target' attribute

Deprecated.

#### 3.3.3. 'type' attribute (mandatory)

Deprecated.

### 3.4. `<list>`

Deprecated. Instead, use `<dl>` for `list/@style "hanging"`; `<ul>` for `list/@style "empty"` or `"symbols"`; and `<ol>` for `list/@style "letters"`, `"numbers"`, `"counter"`, or `"format"`.

This element appears as a child element of: `<t>` (Section 2.53).

Content model:

One or more <t> elements (Section 2.53)

#### 3.4.1. 'counter' attribute

Deprecated. The functionality of this attribute has been replaced with <ol>/@start.

#### 3.4.2. 'hangIndent' attribute

Deprecated. Use <dl> instead.

#### 3.4.3. 'style' attribute

Deprecated.

#### 3.5. <postamble>

Deprecated. Instead, use a regular paragraph after the figure or table.

This element appears as a child element of: <figure> (Section 2.25), and <texttable> (Section 3.8).

Content model:

In any order:

- o Text
- o <bcp14> elements (Section 2.9)
- o <cref> elements (Section 2.16)
- o <em> elements (Section 2.22)
- o <eref> elements (Section 2.24)
- o <iref> elements (Section 2.27)
- o <spanx> elements (Section 3.7)
- o <strong> elements (Section 2.50)
- o <sub> elements (Section 2.51)
- o <sup> elements (Section 2.52)

- o `<tt>` elements (Section 2.62)
- o `<xref>` elements (Section 2.66)

### 3.6. `<preamble>`

Deprecated. Instead, use a regular paragraph before the figure or table.

This element appears as a child element of: `<figure>` (Section 2.25), and `<texttable>` (Section 3.8).

Content model:

In any order:

- o Text
- o `<bcp14>` elements (Section 2.9)
- o `<cref>` elements (Section 2.16)
- o `<em>` elements (Section 2.22)
- o `<eref>` elements (Section 2.24)
- o `<iref>` elements (Section 2.27)
- o `<spanx>` elements (Section 3.7)
- o `<strong>` elements (Section 2.50)
- o `<sub>` elements (Section 2.51)
- o `<sup>` elements (Section 2.52)
- o `<tt>` elements (Section 2.62)
- o `<xref>` elements (Section 2.66)

### 3.7. `<spanx>`

Deprecated.

This element appears as a child element of: `<annotation>` (Section 2.3), `<c>` (Section 3.1), `<postamble>` (Section 3.5), `<preamble>` (Section 3.6), and `<t>` (Section 2.53).

Content model: only text content.

#### 3.7.1. 'style' attribute

Deprecated. Instead of `<spanx style="emph">`, use `<em>`; instead of `<spanx style="strong">`, use `<strong>`; instead of `<spanx style="verb">`, use `<tt>`.

#### 3.7.2. 'xml:space' attribute

Deprecated.

Allowed values:

- o "default"
- o "preserve" (default)

#### 3.8. <texttable>

Deprecated. Use `<table>` instead.

This element appears as a child element of: `<aside>` (Section 2.6), and `<section>` (Section 2.46).

Content model:

In this order:

1. One optional `<name>` element (Section 2.32)
2. One optional `<preamble>` element (Section 3.6)
3. One or more `<ttcol>` elements (Section 3.9)
4. Optional `<c>` elements (Section 3.1)
5. One optional `<postamble>` element (Section 3.5)

#### 3.8.1. 'align' attribute

Deprecated

Allowed values:

- o "left"

- o "center" (default)
- o "right"

#### 3.8.2. 'anchor' attribute

Deprecated

#### 3.8.3. 'style' attribute

Deprecated.

#### 3.8.4. 'suppress-title' attribute

Deprecated.

Allowed values:

- o "true"
- o "false" (default)

#### 3.8.5. 'title' attribute

Deprecated.

#### 3.9. <ttcol>

Deprecated. Instead, use <tr>, <td>, and <th>.

This element appears as a child element of: <texttable> (Section 3.8).

Content model:

In any order:

- o <cref> elements (Section 2.16)
- o <eref> elements (Section 2.24)
- o <iref> elements (Section 2.27)
- o <xref> elements (Section 2.66)
- o Text



### 3.9.1. 'align' attribute

Deprecated.

Allowed values:

- o "left" (default)
- o "center"
- o "right"

### 3.9.2. 'width' attribute

Deprecated.

### 3.10. <vspace>

Deprecated. In earlier versions of this format, <vspace> was often used to get an extra blank line in a list element; in the v3 vocabulary, that can be done instead by using multiple <t> elements inside the <li> element. Other uses have no direct replacement.

This element appears as a child element of: <t> (Section 2.53).

Content model: this element does not have any contents.

### 3.10.1. 'blankLines' attribute

Deprecated.

## 4. Internationalization Considerations

This format is based on [XML], thus does not have any issues representing arbitrary Unicode [UNICODE] characters in text content. The RFC Editor may restrict some of the characters that can be used in a particular RFC; the rules for such restrictions are covered in [NONASCII].

## 5. Security Considerations

The "name" attribute on the <artwork> element (Section 2.5.5) can be used to derive a filename for saving to a local file system. Trusting this kind of information without pre-processing is a known security risk; see Section 4.3 of [RFC6266] for more information.

The "src" attribute on the <artwork> element can be used to read files from the local system. Processing tools must be careful to not

accept dangerous values for the filename, particularly those that contain absolute references outside the current directory.

The "type" attribute of the <artwork> and <sourcecode> elements is meant to encourage formatters to automatically extract known types of content from an RFC or Internet Draft. While extraction is probably safe, those tools might also think that they could further process the extracted content such as by rendering artwork or executing code. Doing so without first sanity-checking the extracted content is clearly a terrible idea from a security perspective. More generally, a tool that is reading XML input needs to be suspicious of any content that it intends to post-process.

All security considerations related to XML processing are relevant as well (see Section 7 of [RFC3470]).

## 6. IANA Considerations

### 6.1. Internet Media Type Registration

IANA maintains the registry of Internet media types [BCP13] at <<https://www.iana.org/assignments/media-types>>.

This document updates the specification for the Internet media type "application/rfc+xml" from the one in [XML2RFCv2]. The following is to be registered with IANA.

Type name: application

Subtype name: rfc+xml

Required parameters: There are no required parameters.

Optional parameters: "charset": This parameter has identical semantics as the charset parameter of the "application/xml" media type specified in Section 9.1 of [RFC7303].

Encoding considerations: Identical to those of "application/xml" as described in Section 9.1 of [RFC7303].

Security considerations: As defined in Section 5. In addition, as this media type uses the "+xml" convention, it inherits the security considerations described in Section 10 of [RFC7303].

Interoperability considerations: Different implementations of this format have had interoperability issues. It is not expected that publication of this application will cause those implementations to be fixed.

Published specification: This specification.

Applications that use this media type: Applications that transform xml2rfc to output representations such as plain text or HTML, plus additional analysis tools.

Fragment identifier considerations: The "anchor" attribute is used for assigning document-wide unique identifiers that can be used as shorthand pointers, as described in Section 2.8 of [XPOINTER].

Additional information:

Deprecated alias names for this type: None.

Magic number(s): As specified for "application/xml" in Section 9.1 of [RFC7303].

File extension(s): .xml or .rfcxml when disambiguation from other XML files is needed

Macintosh file type code(s): TEXT

Person & email address to contact for further information: See Authors Section.

Intended usage: COMMON

Restrictions on usage: None

Author: See Authors Section.

Change controller: RFC Series Editor (rse@rfc-editor.org)

## 6.2. Link Relation Registration

The following is a proposed addition to [LINKRELATIONS].

Relation Name: convertedFrom

Description: The document linked to was later converted to the document that contains this link relation. For example, an RFC can have a link to the Internet Draft that became the RFC; in that case, the link relation would be "convertedFrom".

Reference: This document.

Notes: This relation is different than "predecessor-version" in that "predecessor-version" is for items in a version control system. It

is also different that "previous" in that this relation is used for converted resources, not those that are part of a sequence of resources.

Application Data: none

## 7. Acknowledgments

Thanks to everybody who reviewed this document and provided feedback and/or specification text. Thanks especially go to Julian Reschke for editing [XML2RFCv2] and those who provided feedback on that document.

We also thank Marshall T. Rose for both the original design and the reference implementation of the "xml2rfc" processor.

## 8. References

### 8.1. Normative References

- [XML] Maler, E., Yergeau, F., Paoli, J., Sperberg-McQueen, M., and T. Bray, "Extensible Markup Language (XML) 1.0 (Fifth Edition)", W3C Recommendation REC-xml-20081126, November 2008, <<https://www.w3.org/TR/2008/REC-xml-20081126/>>.
- Latest version available at <<http://www.w3.org/TR/xml>>.
- [XML2RFCv2] Reschke, J., "The 'XML2RFC' version 2 Vocabulary", draft-iab-xml2rfc (work in progress), 2014.

### 8.2. Informative References

- [BCP13] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC 6838, January 2013, <<http://www.rfc-editor.org/info/bcp13>>.
- [DOI] Levine, J., "Assigning Digital Object Identifiers to RFCs", draft-iab-doi (work in progress), 2014.
- [IDGUIDE] Housley, R., "Guidelines to Authors of Internet-Drafts", <<https://www.ietf.org/id-info/guidelines.html>>.
- [LINKRELATIONS] IANA, "Link Relations", <<https://www.iana.org/assignments/link-relations/link-relations.xhtml>>.

- [NONASCII] Flanagan, H., "The Use of Non-ASCII Characters in RFCs", draft-flanagan-nonascii (work in progress), 2015.
- [PREPTOOL] Hoffman, P. and J. Hildebrand, "RFC v3 Prep Tool Description", draft-hoffman-rfcv3-preptool (work in progress), 2015.
- [RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, DOI 10.17487/RFC2026, October 1996, <<http://www.rfc-editor.org/info/rfc2026>>.
- [RFC2397] Masinter, L., "The "data" URL scheme", RFC 2397, DOI 10.17487/RFC2397, August 1998, <<http://www.rfc-editor.org/info/rfc2397>>.
- [RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, DOI 10.17487/RFC2629, June 1999, <<http://www.rfc-editor.org/info/rfc2629>>.
- [RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, DOI 10.17487/RFC3339, July 2002, <<http://www.rfc-editor.org/info/rfc3339>>.
- [RFC3470] Hollenbeck, S., Rose, M., and L. Masinter, "Guidelines for the Use of Extensible Markup Language (XML) within IETF Protocols", BCP 70, RFC 3470, DOI 10.17487/RFC3470, January 2003, <<http://www.rfc-editor.org/info/rfc3470>>.
- [RFC3667] Bradner, S., "IETF Rights in Contributions", RFC 3667, DOI 10.17487/RFC3667, February 2004, <<http://www.rfc-editor.org/info/rfc3667>>.
- [RFC3966] Schulzrinne, H., "The tel URI for Telephone Numbers", RFC 3966, DOI 10.17487/RFC3966, December 2004, <<http://www.rfc-editor.org/info/rfc3966>>.
- [RFC3978] Bradner, S., "IETF Rights in Contributions", RFC 3978, DOI 10.17487/RFC3978, March 2005, <<http://www.rfc-editor.org/info/rfc3978>>.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986,

- January 2005,  
<<http://www.rfc-editor.org/info/rfc3986>>.
- [RFC5741] Daigle, L. and O. Kolkman, "RFC Streams, Headers, and Boilerplates", RFC 5741, DOI 10.17487/RFC5741, December 2009,  
<<http://www.rfc-editor.org/info/rfc5741>>.
- [RFC5743] Falk, A., "Definition of an Internet Research Task Force (IRTF) Document Stream", RFC 5743, DOI 10.17487/RFC5743, December 2009,  
<<http://www.rfc-editor.org/info/rfc5743>>.
- [RFC6068] Duerst, M., Masinter, L., and J. Zawinski, "The 'mailto' URI Scheme", RFC 6068, DOI 10.17487/RFC6068, October 2010,  
<<http://www.rfc-editor.org/info/rfc6068>>.
- [RFC6266] Reschke, J., "Use of the Content-Disposition Header Field in the Hypertext Transfer Protocol (HTTP)", RFC 6266, DOI 10.17487/RFC6266, June 2011,  
<<http://www.rfc-editor.org/info/rfc6266>>.
- [RFC6949] Flanagan, H. and N. Brownlee, "RFC Series Format Requirements and Future Development", RFC 6949, DOI 10.17487/RFC6949, May 2013,  
<<http://www.rfc-editor.org/info/rfc6949>>.
- [RFC7303] Thompson, H. and C. Lilley, "XML Media Types", RFC 7303, DOI 10.17487/RFC7303, July 2014,  
<<http://www.rfc-editor.org/info/rfc7303>>.
- [RFC7322] Heather, H. and S. Ginoza, "RFC Style Guide", RFC 7322, DOI 10.17487/RFC7322, September 2014,  
<<http://www.rfc-editor.org/info/rfc7322>>.
- [RFCPOLICY] RFC Editor, "RFC Editorial Guidelines and Procedures",  
<<https://www.rfc-editor.org/policy.html>>.
- [RNC] Clark, J., "RELAX NG Compact Syntax", OASIS , November 2002, <<https://www.oasis-open.org/committees/relax-ng/compact-20021121.html>>.
- [RNG] ISO/IEC, "Information Technology - Document Schema Definition Languages (DSDL) - Part 2: Regular-Grammar-Based Validation - RELAX NG. Second Edition.", ISO/IEC 19757-2:2008(E), December 2008.

A useful source of RNG-related information is  
<<http://relaxng.org/>>.

- [SVGforRFCs] Brownlee, N., "SVG Drawings for RFCs: SVG 1.2 RFC", draft-brownlee-svg-rfc (work in progress), 2015.
- [TLP2.0] IETF Trust, "Legal Provisions Relating to IETF Documents", February 2009, <<http://trustee.ietf.org/license-info/IETF-TLP-2.htm>>.
- [TLP3.0] IETF Trust, "Legal Provisions Relating to IETF Documents", September 2009, <<http://trustee.ietf.org/license-info/IETF-TLP-3.htm>>.
- [TLP4.0] IETF Trust, "Legal Provisions Relating to IETF Documents", December 2009, <<http://trustee.ietf.org/license-info/IETF-TLP-4.htm>>.
- [UAX24] The Unicode Consortium, "UAX #24: Unicode Script Property", <<http://www.unicode.org/reports/tr24/>>.
- [UNICODE] The Unicode Consortium, "The Unicode Standard".  
  
For the latest version, see  
<<http://www.unicode.org/versions/latest/>>.
- [USASCII] American National Standards Institute, "Coded Character Set -- 7-bit American Standard Code for Information Interchange", ANSI X3.4, 1986.
- [XInclude] Marsh, J., Orchard, D., and D. Veillard, "XML Inclusions (XInclude) Version 1.0 (Second Edition)", W3C Recommendation REC-xinclude-20061115, November 2006, <<http://www.w3.org/TR/xinclude/>>.  
  
Latest version available at  
<<http://www.w3.org/TR/xinclude/>>.
- [XPOINTER] Grosso, P., Maler, E., Marsh, J., and N. Walsh, "XPointer Framework", W3C Recommendation REC-xptr-framework-20030325, March 2003, <<http://www.w3.org/TR/2003/REC-xptr-framework-20030325/>>.  
  
Latest version available at  
<<http://www.w3.org/TR/xptr-framework/>>.

## Appendix A. Front Page ('Boilerplate') Generation

## A.1. The /rfc/@ipr Attribute

This attribute value can take a long list of values, each of which describes an IPR policy for the document (Section 2.45.5). The values are not the result of a grand design, but remain simply for historic reasons. Of these values, only a few are currently in use; all others are supported by various tools for backwards compatibility with old source files.

*\*Note:* some variations of the boilerplate are selected based on the document's date; therefore it is important to specify the "year", "month" and "day" attributes of the <date> element when archiving the XML source of an Internet-Draft on the day of submission.

Disclaimer: THIS ONLY PROVIDES IMPLEMENTATION INFORMATION. IF YOU NEED LEGAL ADVICE, PLEASE CONTACT A LAWYER. For further information, refer to <<http://trustee.ietf.org/docs/IETF-Copyright-FAQ.pdf>>.

For the current "Status of This Memo" text, the submissionType attribute (Section 2.45.12) determines whether a statement about "Code Components" is inserted (which is the case for the value "IETF", which is the default). Other values, such as "independent", suppress this part of the text.

## A.1.1. Current Values: '\*trust200902'

The name for these values refers to the "IETF TRUST Legal Provisions Relating to IETF Documents", sometimes simply called the "TLP", which went into effect on February 15, 2009 ([TLP2.0]). Updates to this document were published on September 12, 2009 ([TLP3.0]) and on December 28, 2009 ([TLP4.0]), modifying the license for code components (see <<http://trustee.ietf.org/license-info/>> for further information). The actual text is located in Section 6 ("Text To Be Included in IETF Documents") of these documents.

The prep tool automatically produces the "correct" text depending on the document's date information (see above):

+-----+-----+   TLP   starting with publication date   +-----+-----+	
[TLP3.0]	2009-11-01
[TLP4.0]	2010-04-01
+-----+-----+	



## A.1.1.1. trust200902

This value should be used unless one of the more specific '\*trust200902' values is a better fit. It produces the text in Sections 6.a and 6.b of the TLP.

## A.1.1.2. noModificationTrust200902

This produces additional text from Section 6.c.i of the TLP:

This document may not be modified, and derivative works of it may not be created, except to format it for publication as an RFC or to translate it into languages other than English.

\*Note:\* this clause is incompatible with RFCs that are published on the Standards Track.

## A.1.1.3. noDerivativesTrust200902

This produces the additional text from Section 6.c.ii of the TLP:

This document may not be modified, and derivative works of it may not be created, and it may not be published except as an Internet-Draft.

\*Note:\* this clause is incompatible with RFCs.

## A.1.1.4. pre5378Trust200902

This produces the additional text from Section 6.c.iii of the TLP, frequently called the "pre-5378 escape clause":

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

See Section 4 of <http://trustee.ietf.org/docs/IETF-Copyright-FAQ.pdf> for further information about when to use this value.

\*Note:\* this text appears under "Copyright Notice", unless the document was published before November 2009, in which case it appears under "Status Of This Memo".

#### A.1.2. Historic Values

##### A.1.2.1. Historic Values: '\*trust200811'

The attribute values "trust200811", "noModificationTrust200811" and "noDerivativesTrust200811" are similar to their "trust200902" counterparts, except that they use text specified in <[http://trustee.ietf.org/license-info/archive/IETF-Trust-License-Policy\\_11-10-08.pdf](http://trustee.ietf.org/license-info/archive/IETF-Trust-License-Policy_11-10-08.pdf)>.

##### A.1.2.2. Historic Values: '\*3978'

The attribute values "full3978", "noModification3978" and "noDerivatives3978" are similar to their counterparts above, except that they use text specified in Section 5 of [RFC3978].

##### A.1.2.3. Historic Values: '\*3667'

The attribute values "full3667", "noModification3667" and "noDerivatives3667" are similar to their counterparts above, except that they use text specified in Section 5 of [RFC3667].

##### A.1.2.4. Historic Values: '\*2026'

The attribute values "full2026" and "noDerivativeWorks2026" are similar to their counterparts above, except that they use text specified in Section 10 of [RFC2026].

The special value "none" was also used back then, and denied the IETF any rights beyond publication as Internet-Draft.

#### A.2. The /rfc/@submissionType Attribute

The RFC Editor publishes documents from different "document streams", of which the "IETF stream" is the most prominent one. Other streams are the "independent stream" (used for things such as administrative information or April 1st RFCs), the "IAB stream" (Internet Architecture Board) and the "IRTF stream" (Internet Research Task Force).

The values for the attribute are "IETF" (the default value), "independent", "IAB", and "IRTF".

Historically, this attribute did not affect the final appearance of

RFCs, except for subtle differences in Copyright notices. Nowadays (as of [RFC5741]), the stream name appears in the first line of the front page, and it also affects the text in the "Status Of This Memo" section.

For current documents, setting `submissionType` attribute will have the following effect:

- o For RFCs, the stream name appears in the upper left corner of the first page (in Internet Drafts, this is either "Network Working Group", or the value of the `<workgroup>` element).
- o For RFCs, it affects the whole "Status Of This Memo" section (see Section 3.2.2 of [RFC5741]).
- o For all RFCs and Internet Drafts, it determines whether the "Copyright Notice" mentions the Copyright on Code Components (see TLP, Section "Text To Be Included in IETF Documents").

#### A.3. The `/rfc/@consensus` Attribute

For some of the publication streams (see Appendix A.2), the "Status Of This Memo" section depends on whether there was a consensus to publish (again, see Section 3.2.2 of [RFC5741]).

The consensus attribute can be used to supply this information. The acceptable values are "true" (the default) and "false"; "yes" and "no" from v2 are deprecated. For documents in the IRTF stream, this value must be "false".

The effect of this value for the various streams is:

- o "independent" and "IAB": none.
- o "IETF": mention that there was an IETF consensus.
- o "IRTF": the text needs to describe the consensus/review as described in Section 2.1 of [RFC5743].

#### Appendix B. The v3 Format and Processing Tools

This section describes topics that are specific to v3 processing tools. Note that there is some discussion of tools in the main body of the document as well. For example, some elements have descriptions of how a processing tool might create output from the element.

The expected design of the tools that will be used with v3 documents

includes:

- o A "prep tool" that takes a v3 document, makes many checks, adds and changes many attribute values, and creates a file that is a "prepared document". The prepared document is a valid v3 document. The prep tool is described in [PREPTOOL].
- o The prep tool is expected to have many modes:
  - \* RFC mode -- The mode used by the RFC Editor to process the input from one of the RFC streams, and to process XML produced during the RFC editing process. The restrictions on the canonical XML for RFCs, as well as how the non-canonical formats will look, are described at <<https://www.rfc-editor.org/rse/wiki/doku.php?id=design:format-and-content-rfcs>>.
  - \* Draft mode -- The mode used by the Internet Draft submission tool. The restrictions for the XML from this mode will be described later.
  - \* Diagnostic mode -- A mode that can be used by document authors to look for errors or warnings before they submit their documents for publication.
  - \* Consolidation mode -- Produces output where no external resources are required to render the file output. This includes expanding the XInclude entities and DTD entities in place, and changing all elements that have "src" attributes with external links into either "data:" URI or content for the element, as specified in [PREPTOOL].
- o Formatting tools that will create HTML, PDF, plain text, and possibly other output formats. These formatters will be created by the IETF, but others can create such tools as well. The IETF tools are expected to take prepared documents as input.

There may also be processing tools that are meant to run on the computers of authors. These tools may be used to produce interim versions of the non-canonical representations so that authors can see how their XML might later be rendered; to create documents in representations different than those supported by the RFC Editor; to possibly create documents that are not meant to be Internet Drafts or RFCs; and to convert XML that has external information into XML that has that external information included.

The prep tool is expected to have clear error reporting, giving more context than just a line number. For example, the error messages

should differentiate between errors in XML and those from the v3 format.

In v2, the grammar was specified as a DTD. In v3, the grammar is specified only as Relax Next Generation (RNG). This means that tools need to work from the RNG, not from a DTD. Some of the features of the v3 grammar cannot be specified as a DTD.

#### B.1. Including External Text with XInclude

All tools for the v3 format are expected to support XInclude [XInclude]. XInclude specifies a processing model and syntax for general purpose inclusion of information that is either on the Internet or local to the user's computer.

In the v3 syntax, XInclude is expressed as the `<xi:include>` element. To use this element, you need to include the "xi" namespace in the `<rfc>` element; that is, you need to specify

```
xmlns:xi="http://www.w3.org/2001/XInclude"
```

as one of the attributes in the `<rfc>` element.

The most common way to use `<xi:include>` is to pull in references that are already formed as XML. Currently, this can be done from `xml2rfc.tools.ietf.org`, but later is expected to be from the RFC Editor. For example, if a document has three normative references, all RFCs, the document might contain:

```
<references>
  <xi:include href="http://xml2rfc.tools.ietf.org/public/rfc/
    bibxml/reference.RFC.2119.xml"/>
  <xi:include href="http://xml2rfc.tools.ietf.org/public/rfc/
    bibxml/reference.RFC.4869.xml"/>
  <xi:include href="http://xml2rfc.tools.ietf.org/public/rfc/
    bibxml/reference.RFC.7169.xml"/>
</references>
```

`<xi:include>` can be used anywhere an XML element could be used (but not where free text is used). For example, if three Internet Drafts are all including a particular paragraph or section verbatim, that text can be kept either in a file or somewhere on the web, and be included with `<xi:include>`. An example of pulling something from the local disk would be:

```
<x:include href="file://home/chris/ietf/drafts/commoncontext.xml"/>
```

In general, XInclude should be used instead of ENTITY references and

XML Processor Instructions (PIs) that allow external inclusions.

## B.2. Anchors and IDs

People writing and reading Internet Drafts and RFCs often want to make reference to specific locations in those documents. In the case of RFC authors, it is common to want to reference another part of their document, such as "see Section 3.2 of this document". Readers, on the other hand, want to reference part of documents that they didn't write, such as "see Section 3.2 of RFC 6949". The XML vocabulary in this document attempts to support both sets of people.

Authors can leave anchors in a document that can later be used for references with the "anchor" attribute. Anchors can be included in the following elements: `<artwork>`, `<aside>`, `<blockquote>`, `<cref>`, `<figure>`, `<li>`, `<reference>`, `<referencegroup>`, `<section>`, `<sourcecode>`, `<t>`, and `<table>`. The author can then refer to that anchor in the "target" attribute of the `<xref>` element.

Readers can refer to any element that has an "anchor" attribute by that attribute. Note, however, that most of the time, elements won't have anchors. In the common case, the reader wants to refer to an element that does not have an "anchor" attribute, but that element has "pn" attribute.

Processing tools add the "pn" attribute to many elements during processing. This attribute and its value are automatically generated by the tool if the attribute is not there; if the attribute is already there, the tool may replace the value.

### B.2.1. Overlapping Values

In the HTML representation of this XML vocabulary, both anchors and "pn" attributes will be used in the "id" attributes of elements. Thus, there can be no overlap between the names entered in "anchor" attributes, in "slugifiedName" attributes, and those that are generated for the "pn" attributes. Also, there are some values for the "anchor" values that are reserved for sections, and those sections can only have those anchor values.

The following rules prevent this overlap:

- o "pn" for regular sections always has the format "s-nnn", where "nnn" is the section or appendix number. For example, this would be "s-2.1.3" for Section 2.1.3 and "s-a" for Appendix A. For the `<abstract>` element, it is always "s-abstract". For the `<note>` element, it is always "s-note-nnn", where "nnn" is a sequential value. For the `<boilerplate>` element, it is always

"s-boilerplate-nnn", where "nnn" is a sequential value.

- o "pn" for <figure> elements always has the format "f-nnn", where "nnn" is the figure number. For example, this would be "f-5" for Figure 5.
- o "pn" for <table> elements always has the format "t-nnn", where "nnn" is the table number. For example, this would be "t-5" for Table 5.
- o "pn" for all elements not listed above always has the format "p-nnn-mmm", where "nnn" is the section number and "mmm" is the relative position in the section. For example, this would be "p-2.1.3-7" for the seventh part number in Section 2.1.3.
- o "slugifiedName" always has the format "n-ttt", where "ttt" is the text of the name after slugification. For example, this would be "n-protocol-overview" for the name "Protocol Overview". The actual conversions done in slugification will be specified at a later time.
- o Anchors must never overlap with any of the above. The easiest way to assure that is to not pick an anchor name that starts with a single letter followed by a hyphen. If an anchor does overlap with one of the types of names above, the processing tool will reject the document.

### B.3. Attributes Controlled by the Prep Tool

Many elements in the v3 vocabulary have new attributes whose role is to hold values generated by the prep tool. These attributes can exist in documents that are input to the prep tool; however, any of these attributes might be added, removed, or changed by the prep tool. Thus, it is explicitly unsafe for a document author to include these attributes and expect that their values will survive processing by the prep tool.

The attributes that are controlled by the prep tool are:

- o The "pn" attribute in any element -- The number for this item within the section. The numbering is shared with other elements of a section. The "pn" attribute is added to many block-level elements inside sections.
- o <artwork> originalSrc -- This attribute is filled with the original value of the "src" attribute if that attribute is removed by the prep tool.

- o `<irefid>` "irefid" attribute -- This attribute is filled with an identifier used when creating indexes.
- o `<name>` "slugifiedName" attribute -- This attribute is filled with a "slugified" version of the text in the element. This attribute can be used in the output formats for elements that have both names and numbers.
- o `<relref>` "derivedLink" attribute -- This attribute is filled with the link that is derived from combining the URI from the reference and the relative part that is either a copy of the "relative" attribute or a section number derived from the "section" attribute.
- o `<relref>` "derivedRemoteContent" attribute -- If the `<relref>` element has text content, this attribute is filled with that content; the "displayFormat" attribute is set to "bare" if that attribute is not already set. If the `<relref>` element has no text content, this attribute is filled with the text for the remote link, such as "Section 2.3" or "Table 5". The prep tool might determine this text by reading the target reference and, if it is a RFC or Internet-Draft in the v3 format, finding the anchor given in the "relative" attribute or derived from the "section" attribute, and using the title of that element. If the reference is not an RFC or Internet-Draft in the v3 format, the text fragment is simply the value of the "relative" or "section" attribute. This will rarely produce a good result in formatted output so, for these documents, the `<relref>` element should contain text content.
- o `<rfc>` "expiresDate" attribute -- This attribute is filled with the date that an Internet Draft expires. The date is in the format yyyy-mm-dd.
- o `<rfc>` "mode" attribute -- This attribute is filled with a string that indicates what mode the prep tool was in when it processed the XML, such as whether it was processing a file to become an Internet-Draft or an RFC.
- o `<rfc>` "scripts" attribute -- This attribute is filled with a list of scripts needed to render this document. The list is comma-separated, with no spaces allowed. The order is unimportant. The names come from [UAX24]. For example, if the document has Chinese characters in it, the value might be "Common, Latin, Han".
- o `<sourcecode>` "originalSrc" attribute -- This attribute is filled with the original value of the "src" attribute if that attribute is removed by the prep tool.



- o <xref> "derivedContent" attribute -- This attribute is filled in if there is no content in the <xref> element. The value for this attribute based on the value in the "displayFormat" attribute. Examples of how this value is filled can be found at Section 2.66.1.

In addition, note that the contents of the <boilerplate> element is controlled by the prep tool.

#### Appendix C. Relax NG Schema

The following is the RelaxNG schema for the v3 format.

```
namespace a = "http://relaxng.org/ns/compatibility/annotations/1.0"
namespace ns1 = "http://www.w3.org/2000/svg"

# xml2rfc Version 3 grammar

rfc =
  element rfc {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute number { text }?,
    [ a:defaultValue = "" ] attribute obsoletes { text }?,
    [ a:defaultValue = "" ] attribute updates { text }?,
    attribute category { text }?,
    attribute mode { text }?,
    [ a:defaultValue = "false" ]
    attribute consensus { "no" | "yes" | "false" | "true" }?,
    attribute seriesNo { text }?,
    attribute ipr {
      "full2026"
      | "noDerivativeWorks2026"
      | "none"
      | "full3667"
      | "noModification3667"
      | "noDerivatives3667"
      | "full3978"
      | "noModification3978"
      | "noDerivatives3978"
      | "trust200811"
      | "noModificationTrust200811"
      | "noDerivativesTrust200811"
      | "trust200902"
      | "noModificationTrust200902"
      | "noDerivativesTrust200902"
      | "pre5378Trust200902"
```

```
    }?,
    attribute iprExtract { xsd:IDREF }?,
    [ a:defaultValue = "IETF" ]
    attribute submissionType {
      "IETF" | "IAB" | "IRTF" | "independent"
    }?,
    attribute docName { text }?,
    [ a:defaultValue = "false" ]
    attribute sortRefs { "true" | "false" }?,
    [ a:defaultValue = "true" ]
    attribute symRefs { "true" | "false" }?,
    [ a:defaultValue = "true" ]
    attribute tocInclude { "true" | "false" }?,
    [ a:defaultValue = "3" ] attribute tocDepth { text }?,
    attribute prepTime { text }?,
    [ a:defaultValue = "true" ]
    attribute indexInclude { "true" | "false" }?,
    attribute version { text }?,
    [ a:defaultValue = "Common, Latin" ] attribute scripts { text }?,
    attribute expiresDate { text }?,
    link*,
    front,
    middle,
    back?
  }

link =
  element link {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute href { text },
    attribute rel { text }?
  }

front =
  element front {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    title,
    author+,
    date?,
    area*,
    workgroup*,
    keyword*,
    abstract?,
    seriesInfo*,
    note*,
    boilerplate?
```

```
    }

    title =
      element title {
        attribute xml:base { text }?,
        attribute xml:lang { text }?,
        attribute abbrev { text }?,
        attribute ascii { text }?,
        text
      }

    author =
      element author {
        attribute xml:base { text }?,
        attribute xml:lang { text }?,
        attribute initials { text }?,
        attribute asciiInitials { text }?,
        attribute surname { text }?,
        attribute asciiSurname { text }?,
        attribute fullname { text }?,
        attribute role { "editor" }?,
        attribute asciiFullname { text }?,
        organization?,
        address?
      }

    organization =
      element organization {
        attribute xml:base { text }?,
        attribute xml:lang { text }?,
        attribute abbrev { text }?,
        attribute ascii { text }?,
        text
      }

    address =
      element address {
        attribute xml:base { text }?,
        attribute xml:lang { text }?,
        postal?,
        phone?,
        facsimile?,
        email?,
        uri?
      }

    postal =
      element postal {
```

```
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    ((city | code | country | region | street)* | postalLine+)
  }

street =
  element street {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute ascii { text }?,
    text
  }

city =
  element city {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute ascii { text }?,
    text
  }

region =
  element region {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute ascii { text }?,
    text
  }

code =
  element code {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute ascii { text }?,
    text
  }

country =
  element country {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute ascii { text }?,
    text
  }

postalLine =
  element postalLine {
    attribute xml:base { text }?,
```

```
    attribute xml:lang { text }?,
    attribute ascii { text }?,
    text
  }

phone =
  element phone {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    text
  }

facsimile =
  element facsimile {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    text
  }

email =
  element email {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute ascii { text }?,
    text
  }

uri =
  element uri {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    text
  }

date =
  element date {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute day { text }?,
    attribute month { text }?,
    attribute year { text }?,
    empty
  }

area =
  element area {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
```

```
    text
  }

workgroup =
  element workgroup {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    text
  }

keyword =
  element keyword {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    text
  }

abstract =
  element abstract {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    (dl | ol | t | ul)+
  }

note =
  element note {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute title { text }?,
    attribute pn { text }?,
    [ a:defaultValue = "false" ]
    attribute removeInRFC { "true" | "false" }?,
    name?,
    (dl | ol | t | ul)+
  }

boilerplate =
  element boilerplate {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute pn { text }?,
    section+
  }

middle =
  element middle {
```

```
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    section+
  }

section =
  element section {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    attribute title { text }?,
    [ a:defaultValue = "true" ]
    attribute numbered { "true" | "false" }?,
    [ a:defaultValue = "default" ]
    attribute toc { "include" | "exclude" | "default" }?,
    [ a:defaultValue = "false" ]
    attribute removeInRFC { "true" | "false" }?,
    name?,
    (artwork
    | aside
    | blockquote
    | dl
    | figure
    | iref
    | ol
    | sourcecode
    | t
    | table
    | texttable
    | ul)*,
    section*
  }

name =
  element name {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute slugifiedName { text }?,
    (text | cref | eref | relref | tt | xref)*
  }

t =
  element t {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
```

```
    attribute hangText { text }?,
    [ a:defaultValue = "false" ]
    attribute keepWithNext { "false" | "true" }?,
    [ a:defaultValue = "false" ]
    attribute keepWithPrevious { "false" | "true" }?,
    (text
      | bcp14
      | cref
      | em
      |eref
      |iref
      |\list
      |relref
      |spanx
      |strong
      |sub
      |sup
      |tt
      |vspace
      |xref)*
  }

aside =
  element aside {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    (artwork | dl | figure | iref | \list | ol | t | table | ul)*
  }

blockquote =
  element blockquote {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    attribute cite { text }?,
    attribute quotedFrom { text }?,
    ((artwork | dl | figure | ol | sourcecode | t | ul)+
      | (text
        | bcp14
        | cref
        | em
        |eref
        |iref
        |relref
        |strong
```



```
        | sub
        | sup
        | tt
        | xref)+)
    }

\list =
  element list {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    [ a:defaultValue = "empty" ] attribute style { text }?,
    attribute hangIndent { text }?,
    attribute counter { text }?,
    t+
  }

ol =
  element ol {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    [ a:defaultValue = "1" ] attribute type { text }?,
    [ a:defaultValue = "1" ] attribute start { text }?,
    attribute group { text }?,
    [ a:defaultValue = "normal" ]
    attribute spacing { "normal" | "compact" }?,
    li+
  }

ul =
  element ul {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    [ a:defaultValue = "normal" ]
    attribute spacing { "normal" | "compact" }?,
    [ a:defaultValue = "false" ]
    attribute empty { "false" | "true" }?,
    li+
  }

li =
  element li {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
```

```
    attribute pn { text }?,
    ((artwork | dl | figure | ol | sourcecode | t | ul)+
     | (text
       | bcp14
       | cref
       | em
       | eref
       | iref
       | relref
       | strong
       | sub
       | sup
       | tt
       | xref)+)
  }

dl =
  element dl {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    [ a:defaultValue = "normal" ]
    attribute spacing { "normal" | "compact" }?,
    [ a:defaultValue = "true" ]
    attribute hanging { "false" | "true" }?,
    (dt, dd)+
  }

dt =
  element dt {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    (text
     | bcp14
     | cref
     | em
     | eref
     | iref
     | relref
     | strong
     | sub
     | sup
     | tt
     | xref)*
  }
```

```
dd =
  element dd {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    ((artwork | dl | figure | ol | sourcecode | t | ul)+
    | (text
      | bcp14
      | cref
      | em
      | eref
      | iref
      | relref
      | strong
      | sub
      | sup
      | tt
      | xref)+)
  }

xref =
  element xref {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute target { xsd:IDREF },
    [ a:defaultValue = "false" ]
    attribute pageno { "true" | "false" }?,
    [ a:defaultValue = "default" ]
    attribute format { "default" | "title" | "counter" | "none" }?,
    attribute derivedContent { text }?,
    text
  }

relref =
  element relref {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute target { xsd:IDREF },
    [ a:defaultValue = "of" ]
    attribute displayFormat { "of" | "comma" | "parens" | "bare" }?,
    (attribute section { text },
    attribute relative { text }?)?,
    attribute derivedRemoteContent { text }?,
    attribute derivedLink { text }?,
    text
  }
```

```
eref =
  element eref {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute target { text },
    text
  }

iref =
  element iref {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute item { text },
    [ a:defaultValue = "" ] attribute subitem { text }?,
    [ a:defaultValue = "false" ]
    attribute primary { "true" | "false" }?,
    [ a:defaultValue = "" ] attribute irefid { text }?,
    empty
  }

cref =
  element cref {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute source { text }?,
    [ a:defaultValue = "true" ]
    attribute display { "true" | "false" }?,
    (text | em | eref | relref | strong | sub | sup | tt | xref)*
  }

tt =
  element tt {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text
     | bcp14
     | cref
     | em
     | eref
     | iref
     | relref
     | strong
     | sub
     | sup
     | xref)*
  }
```

```
strong =
  element strong {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text
      | bcp14
      | cref
      | em
      | eref
      | iref
      | relref
      | sub
      | sup
      | tt
      | xref)*
  }
```

```
em =
  element em {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text
      | bcp14
      | cref
      | eref
      | iref
      | relref
      | strong
      | sub
      | sup
      | tt
      | xref)*
  }
```

```
sub =
  element sub {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text
      | bcp14
      | cref
      | em
      | eref
      | iref
      | relref
      | strong
      | tt
      | xref)*
  }
```

```
    }

sup =
  element sup {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text
      | bcp14
      | cref
      | em
      | eref
      | iref
      | relref
      | strong
      | tt
      | xref)*
  }

spanx =
  element spanx {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    [ a:defaultValue = "preserve" ]
    attribute xml:space { "default" | "preserve" }?,
    [ a:defaultValue = "emph" ] attribute style { text }?,
    text
  }

vspace =
  element vspace {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    [ a:defaultValue = "0" ] attribute blankLines { text }?,
    empty
  }

figure =
  element figure {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    [ a:defaultValue = "" ] attribute title { text }?,
    [ a:defaultValue = "false" ]
    attribute suppress-title { "true" | "false" }?,
    attribute src { text }?,
    [ a:defaultValue = "left" ]
    attribute align { "left" | "center" | "right" }?,
```

```
[ a:defaultValue = "" ] attribute alt { text }?,
[ a:defaultValue = "" ] attribute width { text }?,
[ a:defaultValue = "" ] attribute height { text }?,
name?,
iref*,
preamble?,
(artwork | sourcecode)+,
postamble?
}

table =
  element table {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    name?,
    iref*,
    thead?,
    tbody+,
    tfoot?
  }

preamble =
  element preamble {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text
      | bcp14
      | cref
      | em
      | eref
      | iref
      | relref
      | spanx
      | strong
      | sub
      | sup
      | tt
      | xref)*
  }

artwork =
  element artwork {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
```

```
    attribute xml:space { text }?,
    [ a:defaultValue = "" ] attribute name { text }?,
    [ a:defaultValue = "" ] attribute type { text }?,
    attribute src { text }?,
    [ a:defaultValue = "left" ]
    attribute align { "left" | "center" | "right" }?,
    [ a:defaultValue = "" ] attribute alt { text }?,
    [ a:defaultValue = "" ] attribute width { text }?,
    [ a:defaultValue = "" ] attribute height { text }?,
    attribute originalSrc { text }?,
    (text* | svg)
  }

svg =
  element ns1:svg {
    [ a:defaultValue = "can be found in draft-brownlee-svg-rfc" ]
    attribute TheRealDefinition { text }*
  }

sourcecode =
  element sourcecode {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    [ a:defaultValue = "" ] attribute name { text }?,
    [ a:defaultValue = "" ] attribute type { text }?,
    attribute src { text }?,
    attribute originalSrc { text }?,
    text
  }

thead =
  element thead {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    tr+
  }

tbody =
  element tbody {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    tr+
```



```
    }

tfoot =
  element tfoot {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    tr+
  }

tr =
  element tr {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    [ a:defaultValue = "0" ] attribute border { text }?,
    (td | th)+
  }

td =
  element td {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    [ a:defaultValue = "0" ] attribute border { text }?,
    [ a:defaultValue = "0" ] attribute colspan { text }?,
    [ a:defaultValue = "0" ] attribute rowspan { text }?,
    [ a:defaultValue = "left" ]
    attribute align { "left" | "center" | "right" }?,
    (t+
    | (text
      | artwork
      | bcpl4
      | br
      | cref
      | dl
      | em
      | eref
      | figure
      | iref
      | ol
      | relref
      | sourcecode
      | strong
      | sub
      | sup
```

```

        | tt
        | ul
        | xref)+)
    }

th =
  element th {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute pn { text }?,
    [ a:defaultValue = "0" ] attribute border { text }?,
    [ a:defaultValue = "0" ] attribute colspan { text }?,
    [ a:defaultValue = "0" ] attribute rowspan { text }?,
    [ a:defaultValue = "left" ]
    attribute align { "left" | "center" | "right" }?,
    (t+
      | (text
        | artwork
        | bcpl4
        | br
        | cref
        | dl
        | em
        | eref
        | figure
        | iref
        | ol
        | relref
        | sourcecode
        | strong
        | sub
        | sup
        | tt
        | ul
        | xref)+)
    )
  }

postamble =
  element postamble {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text | cref | eref | iref | spanx | xref)*
  }

texttable =
  element texttable {
    attribute xml:base { text }?,

```

```
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    [ a:defaultValue = "" ] attribute title { text }?,
    [ a:defaultValue = "false" ]
    attribute suppress-title { "true" | "false" }?,
    [ a:defaultValue = "center" ]
    attribute align { "left" | "center" | "right" }?,
    [ a:defaultValue = "full" ]
    attribute style { "all" | "none" | "headers" | "full" }?,
    name?,
    preamble?,
    ttcol+,
    c*,
    postamble?
  }

ttcol =
  element ttcol {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute width { text }?,
    [ a:defaultValue = "left" ]
    attribute align { "left" | "center" | "right" }?,
    (cref | eref | iref | xref | text)*
  }

c =
  element c {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text | cref | eref | iref | spanx | xref)*
  }

bcp14 =
  element bcp14 {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    text
  }

br =
  element br {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    empty
  }

back =
```

```
element back {
  attribute xml:base { text }?,
  attribute xml:lang { text }?,
  displayreference*,
  references*,
  section*
}

displayreference =
  element displayreference {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute target { xsd:IDREF },
    attribute to { text }
  }

references =
  element references {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID }?,
    attribute title { text }?,
    name?,
    (reference | referencegroup)*
  }

reference =
  element reference {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID },
    attribute target { text }?,
    [ a:defaultValue = "true" ]
    attribute quoteTitle { "true" | "false" }?,
    front,
    (annotation | format | refcontent | seriesInfo)*
  }

referencegroup =
  element referencegroup {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute anchor { xsd:ID },
    reference+
  }

seriesInfo =
  element seriesInfo {
```

```
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute name { text },
    attribute value { text },
    attribute asciiName { text }?,
    attribute asciiValue { text }?,
    empty
  }

format =
  element format {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    attribute target { text }?,
    attribute type { text },
    attribute octets { text }?,
    empty
  }

annotation =
  element annotation {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text
      | bcp14
      | cref
      | em
      | eref
      | iref
      | relref
      | spanx
      | strong
      | sub
      | sup
      | tt
      | xref)*
  }

refcontent =
  element refcontent {
    attribute xml:base { text }?,
    attribute xml:lang { text }?,
    (text | bcp14 | em | strong | sub | sup | tt)*
  }

start = rfc
```

## Appendix D. Schema Differences from v2

The following is a non-normative comparison of the v3 format to the v2 format. A "-" indicates lines removed from the v2 schema, and a "+" indicates lines added to the v3 schema.

```
namespace a =
"http://relaxng.org/ns/compatibility/annotations/1.0"
+ namespace ns1 = "http://www.w3.org/2000/svg"

+ # xml2rfc Version 3 grammar
rfc =
  element rfc {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
    attribute number { text }?,
    [ a:defaultValue = "" ] attribute obsoletes { text }?,
    [ a:defaultValue = "" ] attribute updates { text }?,
-   attribute category { "std" | "bcp" | "info" | "exp" |
-   "historic" }?,
-   attribute consensus { "no" | "yes" }?,
+   attribute category { text }?,
+   attribute mode { text }?,
+   [ a:defaultValue = "false" ]
+   attribute consensus { "no" | "yes" | "false" | "true" }?,
+   attribute seriesNo { text }?,
+   attribute ipr {
      "full2026"
      | "noDerivativeWorks2026"
      | "none"
      | "full3667"
      | "noModification3667"
      | "noDerivatives3667"
      | "full3978"
      | "noModification3978"
      | "noDerivatives3978"
      | "trust200811"
      | "noModificationTrust200811"
      | "noDerivativesTrust200811"
      | "trust200902"
      | "noModificationTrust200902"
      | "noDerivativesTrust200902"
      | "pre5378Trust200902"
    }?,
+   attribute iprExtract { xsd:IDREF }?,
    [ a:defaultValue = "IETF" ]
+   attribute submissionType {
```

```

        "IETF" | "IAB" | "IRTF" | "independent"
    }?,
    attribute docName { text }?,
-   [ a:defaultValue = "en" ] attribute xml:lang { text }?,
+   [ a:defaultValue = "false" ]
+   attribute sortRefs { "true" | "false" }?,
+   [ a:defaultValue = "true" ]
+   attribute symRefs { "true" | "false" }?,
+   [ a:defaultValue = "true" ]
+   attribute tocInclude { "true" | "false" }?,
+   [ a:defaultValue = "3" ] attribute tocDepth { text }?,
+   attribute prepTime { text }?,
+   [ a:defaultValue = "true" ]
+   attribute indexInclude { "true" | "false" }?,
+   attribute version { text }?,
+   [ a:defaultValue = "Common, Latin" ] attribute scripts { text
+ }?,
+   attribute expiresDate { text }?,
+   link*,
+   front,
+   middle,
+   back?
    }
+ link =
+   element link {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute href { text },
+     attribute rel { text }?
+   }
+   front =
+     element front {
-       title, author+, date, area*, workgroup*, keyword*, abstract?,
-       note*
+       attribute xml:base { text }?,
+       attribute xml:lang { text }?,
+       title,
+       author+,
+       date?,
+       area*,
+       workgroup*,
+       keyword*,
+       abstract?,
+       seriesInfo*,
+       note*,
+       boilerplate?
+     }
+   title =

```

```
    element title {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute abbrev { text }?,
+   attribute ascii { text }?,
+   text
    }
    author =
    element author {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute initials { text }?,
+   attribute asciiInitials { text }?,
+   attribute surname { text }?,
+   attribute asciiSurname { text }?,
+   attribute fullname { text }?,
+   attribute role { "editor" }?,
+   attribute asciiFullname { text }?,
+   organization?,
+   address?
    }
    organization =
    element organization {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute abbrev { text }?,
+   attribute ascii { text }?,
+   text
    }
+   address =
+   element address {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   postal?,
+   phone?,
+   facsimile?,
+   email?,
+   uri?
+   }
+   postal =
+   element postal {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   ((city | code | country | region | street)* | postalLine+)
+   }
+   street =
+   element street {
+   attribute xml:base { text }?,
```



```
+   attribute xml:lang { text }?,
+   attribute ascii { text }?,
+   text
+ }
+ city =
+   element city {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute ascii { text }?,
+     text
+   }
+ region =
+   element region {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute ascii { text }?,
+     text
+   }
+ code =
+   element code {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute ascii { text }?,
+     text
+   }
+ country =
+   element country {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute ascii { text }?,
+     text
+   }
+ postalLine =
+   element postalLine {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute ascii { text }?,
+     text
+   }
+ phone =
+   element phone {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     text
+   }
+ facsimile =
+   element facsimile {
+     attribute xml:base { text }?,
```

```
+   attribute xml:lang { text }?,
+   text
+ }
+ email =
+   element email {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute ascii { text }?,
+     text
+   }
+ uri =
+   element uri {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     text
+   }
- address = element address { postal?, phone?, facsimile?, email?,
- uri? }
- postal = element postal { street+, (city | region | code |
- country)* }
- street = element street { text }
- city = element city { text }
- region = element region { text }
- code = element code { text }
- country = element country { text }
- phone = element phone { text }
- facsimile = element facsimile { text }
- email = element email { text }
- uri = element uri { text }
  date =
    element date {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute day { text }?,
+     attribute month { text }?,
+     attribute year { text }?,
+     empty
    }
- area = element area { text }
- workgroup = element workgroup { text }
- keyword = element keyword { text }
- abstract = element abstract { t+ }
+ area =
+   element area {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     text
+   }
```

```
+ workgroup =
+   element workgroup {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     text
+   }
+ keyword =
+   element keyword {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     text
+   }
+ abstract =
+   element abstract {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     (dl | ol | t | ul)+
+   }
+ note =
+   element note {
+     attribute title { text },
+     t+
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute title { text }?,
+     attribute pn { text }?,
+     [ a:defaultValue = "false" ]
+     attribute removeInRFC { "true" | "false" }?,
+     name?,
+     (dl | ol | t | ul)+
+   }
+ boilerplate =
+   element boilerplate {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute pn { text }?,
+     section+
+   }
+ middle =
+   element middle {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     section+
+   }
- middle = element middle { section+ }
section =
```

```

    element section {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute anchor { xsd:ID }?,
-   attribute title { text },
+   attribute pn { text }?,
+   attribute title { text }?,
+   [ a:defaultValue = "true" ]
+   attribute numbered { "true" | "false" }?,
+   [ a:defaultValue = "default" ]
+   attribute toc { "include" | "exclude" | "default" }?,
-   (t | figure | texttable | iref)*,
+   [ a:defaultValue = "false" ]
+   attribute removeInRFC { "true" | "false" }?,
+   name?,
+   (artwork
+   | aside
+   | blockquote
+   | dl
+   | figure
+   | iref
+   | ol
+   | sourcecode
+   | t
+   | table
+   | texttable
+   | ul)*,
    section*
  }
+ name =
+   element name {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute slugifiedName { text }?,
+   (text | cref | eref | relref | tt | xref)*
+   }
  t =
    element t {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute anchor { xsd:ID }?,
+   attribute pn { text }?,
+   attribute hangText { text }?,
+   [ a:defaultValue = "false" ]
+   attribute keepWithNext { "false" | "true" }?,
+   [ a:defaultValue = "false" ]
+   attribute keepWithPrevious { "false" | "true" }?,
+   (text

```

```

-      | \list
-      | figure
-      | xref
+      | bcpl4
+      | cref
+      | em
      | eref
      | iref
-      | cref
+      | \list
+      | relref
+      | spanx
-      | vspace)*
+      | strong
+      | sub
+      | sup
+      | tt
+      | vspace
+      | xref)*
+    }
+  aside =
+    element aside {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
+      attribute anchor { xsd:ID }?,
+      attribute pn { text }?,
+      (artwork | dl | figure | iref | \list | ol | t | table | ul)*
+    }
+  blockquote =
+    element blockquote {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
+      attribute anchor { xsd:ID }?,
+      attribute pn { text }?,
+      attribute cite { text }?,
+      attribute quotedFrom { text }?,
+      ((artwork | dl | figure | ol | sourcecode | t | ul)+
+      | (text
+      | bcpl4
+      | cref
+      | em
+      | eref
+      | iref
+      | relref
+      | strong
+      | sub
+      | sup
+      | tt

```

```

+         | xref)+)
+     }
+ \list =
+   element list {
+     attribute style { text }?,
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     [ a:defaultValue = "empty" ] attribute style { text }?,
+     attribute hangIndent { text }?,
+     attribute counter { text }?,
+     t+
+   }
+ ol =
+   element ol {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     [ a:defaultValue = "1" ] attribute type { text }?,
+     [ a:defaultValue = "1" ] attribute start { text }?,
+     attribute group { text }?,
+     [ a:defaultValue = "normal" ]
+     attribute spacing { "normal" | "compact" }?,
+     li+
+   }
+ ul =
+   element ul {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     [ a:defaultValue = "normal" ]
+     attribute spacing { "normal" | "compact" }?,
+     [ a:defaultValue = "false" ]
+     attribute empty { "false" | "true" }?,
+     li+
+   }
+ li =
+   element li {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     ((artwork | dl | figure | ol | sourcecode | t | ul)+
+     | (text
+       | bcp14
+       | cref
+       | em

```

```

+         |   eref
+         |   iref
+         |   relref
+         |   strong
+         |   sub
+         |   sup
+         |   tt
+         |   xref)+)
+     }
+ dl =
+   element dl {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     [ a:defaultValue = "normal" ]
+     attribute spacing { "normal" | "compact" }?,
+     [ a:defaultValue = "true" ]
+     attribute hanging { "false" | "true" }?,
+     (dt, dd)+
+   }
+ dt =
+   element dt {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     (text
+       | bcp14
+       | cref
+       | em
+       | eref
+       | iref
+       | relref
+       | strong
+       | sub
+       | sup
+       | tt
+       | xref)*
+     )
+   }
+ dd =
+   element dd {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     ((artwork | dl | figure | ol | sourcecode | t | ul)+
+      | (text

```

```

+         | bcp14
+         | cref
+         | em
+         | eref
+         | iref
+         | relref
+         | strong
+         | sub
+         | sup
+         | tt
+         | xref)+)
+     }
+     xref =
+       element xref {
+         attribute xml:base { text }?,
+         attribute xml:lang { text }?,
+         attribute target { xsd:IDREF },
+         [ a:defaultValue = "false" ] attribute pageno { "true" |
-         "false" }?,
+         [ a:defaultValue = "false" ]
+         attribute pageno { "true" | "false" }?,
+         [ a:defaultValue = "default" ]
+         attribute format { "counter" | "title" | "none" | "default"
-         attribute format { "default" | "title" | "counter" | "none"
+ }?,
+         attribute derivedContent { text }?,
+         text
+       }
+     relref =
+       element relref {
+         attribute xml:base { text }?,
+         attribute xml:lang { text }?,
+         attribute target { xsd:IDREF },
+         [ a:defaultValue = "of" ]
+         attribute displayFormat { "of" | "comma" | "parens" | "bare"
+ }?,
+         (attribute section { text },
+         attribute relative { text }?)?,
+         attribute derivedRemoteContent { text }?,
+         attribute derivedLink { text }?,
+         text
+       }
+     eref =
+       element eref {
+         attribute xml:base { text }?,
+         attribute xml:lang { text }?,
+         attribute target { text },
+         text

```



```

    }
    iref =
      element iref {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
        attribute item { text },
        [ a:defaultValue = "" ] attribute subitem { text }?,
        [ a:defaultValue = "false" ]
        attribute primary { "true" | "false" }?,
+      [ a:defaultValue = "" ] attribute irefid { text }?,
        empty
      }
    cref =
      element cref {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
        attribute anchor { xsd:ID }?,
        attribute source { text }?,
-      text
+      [ a:defaultValue = "true" ]
+      attribute display { "true" | "false" }?,
+      (text | em | eref | relref | strong | sub | sup | tt | xref)*
+    }
+  tt =
+    element tt {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
+      (text
+      | bcp14
+      | cref
+      | em
+      | eref
+      | iref
+      | relref
+      | strong
+      | sub
+      | sup
+      | xref)*
+    }
+  strong =
+    element strong {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
+      (text
+      | bcp14
+      | cref
+      | em
+      | eref

```

```
+      | iref
+      | relref
+      | sub
+      | sup
+      | tt
+      | xref)*
+    }
+ em =
+   element em {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     (text
+      | bcp14
+      | cref
+      | eref
+      | iref
+      | relref
+      | strong
+      | sub
+      | sup
+      | tt
+      | xref)*
+    )
+   sub =
+    element sub {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
+      (text
+       | bcp14
+       | cref
+       | em
+       | eref
+       | iref
+       | relref
+       | strong
+       | tt
+       | xref)*
+     )
+   sup =
+    element sup {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
+      (text
+       | bcp14
+       | cref
+       | em
+       | eref
+       | iref
```

```

+      | relref
+      | strong
+      | tt
+      | xref)*
    }
    spanx =
      element spanx {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
        [ a:defaultValue = "preserve" ]
        attribute xml:space { "default" | "preserve" }?,
        [ a:defaultValue = "emph" ] attribute style { text }?,
        text
      }
    vspace =
      element vspace {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
        [ a:defaultValue = "0" ] attribute blankLines { text }?,
        empty
      }
    figure =
      element figure {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
        attribute anchor { xsd:ID }?,
+      attribute pn { text }?,
        [ a:defaultValue = "" ] attribute title { text }?,
        [ a:defaultValue = "false" ]
        attribute suppress-title { "true" | "false" }?,
        attribute src { text }?,
        [ a:defaultValue = "left" ]
        attribute align { "left" | "center" | "right" }?,
        [ a:defaultValue = "" ] attribute alt { text }?,
        [ a:defaultValue = "" ] attribute width { text }?,
        [ a:defaultValue = "" ] attribute height { text }?,
+      name?,
        iref*,
        preamble?,
-      artwork,
+      (artwork | sourcecode)+,
        postamble?
      }
+ table =
+   element table {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,

```

```

+   attribute pn { text }?,
+   name?,
+   iref*,
+   thead?,
+   tbody+,
+   tfoot?
+ }
+ preamble =
-   element preamble { (text | xref | eref | iref | cref | spanx)* }
+   element preamble {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     (text
+       | bcp14
+       | cref
+       | em
+       | eref
+       | iref
+       | relref
+       | spanx
+       | strong
+       | sub
+       | sup
+       | tt
+       | xref)*
+   )
+   }
+   artwork =
-   element artwork {
-     [ a:defaultValue = "preserve" ]
-     attribute xml:space { "default" | "preserve" }?,
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     attribute xml:space { text }?,
+     [ a:defaultValue = "" ] attribute name { text }?,
+     [ a:defaultValue = "" ] attribute type { text }?,
+     attribute src { text }?,
+     [ a:defaultValue = "left" ]
+     attribute align { "left" | "center" | "right" }?,
+     [ a:defaultValue = "" ] attribute alt { text }?,
+     [ a:defaultValue = "" ] attribute width { text }?,
+     [ a:defaultValue = "" ] attribute height { text }?,
-     text*
+     attribute originalSrc { text }?,
+     (text* | svg)
+   }
+   svg =

```

```
+   element ns1:svg {
+     [ a:defaultValue = "can be found in draft-brownlee-svg-rfc" ]
+     attribute TheRealDefinition { text }*
+   }
+ sourcecode =
+   element sourcecode {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     [ a:defaultValue = "" ] attribute name { text }?,
+     [ a:defaultValue = "" ] attribute type { text }?,
+     attribute src { text }?,
+     attribute originalSrc { text }?,
+     text
+   }
+ thead =
+   element thead {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     tr+
+   }
+ tbody =
+   element tbody {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     tr+
+   }
+ tfoot =
+   element tfoot {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     tr+
+   }
+ tr =
+   element tr {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     [ a:defaultValue = "0" ] attribute border { text }?,
+     (td | th)+
  }
```

```

+   }
+ td =
+   element td {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     [ a:defaultValue = "0" ] attribute border { text }?,
+     [ a:defaultValue = "0" ] attribute colspan { text }?,
+     [ a:defaultValue = "0" ] attribute rowspan { text }?,
+     [ a:defaultValue = "left" ]
+     attribute align { "left" | "center" | "right" }?,
+     (t+
+     | (text
+       | artwork
+       | bcp14
+       | br
+       | cref
+       | dl
+       | em
+       | eref
+       | figure
+       | iref
+       | ol
+       | relref
+       | sourcecode
+       | strong
+       | sub
+       | sup
+       | tt
+       | ul
+       | xref)+)
+   }
+ th =
+   element th {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute pn { text }?,
+     [ a:defaultValue = "0" ] attribute border { text }?,
+     [ a:defaultValue = "0" ] attribute colspan { text }?,
+     [ a:defaultValue = "0" ] attribute rowspan { text }?,
+     [ a:defaultValue = "left" ]
+     attribute align { "left" | "center" | "right" }?,
+     (t+
+     | (text
+       | artwork
+       | bcp14
+       | br

```

```

+         | cref
+         | dl
+         | em
+         | eref
+         | figure
+         | iref
+         | ol
+         | relref
+         | sourcecode
+         | strong
+         | sub
+         | sup
+         | tt
+         | ul
+         | xref)+)
    }
    postamble =
-   element postamble { (text | xref | eref | iref | cref | spanx)*
+   element postamble {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     (text | cref | eref | iref | spanx | xref)*
+   }
    texttable =
      element texttable {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
      attribute anchor { xsd:ID }?,
      [ a:defaultValue = "" ] attribute title { text }?,
      [ a:defaultValue = "false" ]
      attribute suppress-title { "true" | "false" }?,
      [ a:defaultValue = "center" ]
      attribute align { "left" | "center" | "right" }?,
      [ a:defaultValue = "full" ]
      attribute style { "all" | "none" | "headers" | "full" }?,
+     name?,
      preamble?,
      ttcol+,
      c*,
      postamble?
    }
    ttcol =
      element ttcol {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
      attribute width { text }?,
      [ a:defaultValue = "left" ]
      attribute align { "left" | "center" | "right" }?,

```

```

+   (cref | eref | iref | xref | text)*
+ }
+ c =
+   element c {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     (text | cref | eref | iref | spanx | xref)*
+   }
+ bcp14 =
+   element bcp14 {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     text
+   }
- c = element c { (text | xref | eref | iref | cref | spanx)* }
- back = element back { references*, section* }
+ br =
+   element br {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     empty
+   }
+ back =
+   element back {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     displayreference*,
+     references*,
+     section*
+   }
+ displayreference =
+   element displayreference {
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute target { xsd:IDREF },
+     attribute to { text }
+   }
+ references =
+   element references {
-   [ a:defaultValue = "References" ] attribute title { text }?,
-   reference+
+     attribute xml:base { text }?,
+     attribute xml:lang { text }?,
+     attribute anchor { xsd:ID }?,
+     attribute title { text }?,
+     name?,
+     (reference | referencegroup)*
+   }

```



```

reference =
  element reference {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute anchor { xsd:ID },
+   attribute target { text }?,
+   [ a:defaultValue = "true" ]
+   attribute quoteTitle { "true" | "false" }?,
+   front,
-   seriesInfo*,
-   format*,
-   annotation*
+   (annotation | format | refcontent | seriesInfo)*
+ }
+ referencegroup =
+   element referencegroup {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute anchor { xsd:ID },
+   reference+
+ }
seriesInfo =
  element seriesInfo {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute name { text },
+   attribute value { text },
+   attribute asciiName { text }?,
+   attribute asciiValue { text }?,
+   empty
  }
format =
  element format {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   attribute target { text }?,
+   attribute type { text },
+   attribute octets { text }?,
+   empty
  }
annotation =
-   element annotation { (text | xref | eref | iref | cref |
-   spanx)* }
+   element annotation {
+   attribute xml:base { text }?,
+   attribute xml:lang { text }?,
+   (text
+   | bcp14

```

```

+      | cref
+      | em
+      | eref
+      | iref
+      | relref
+      | spanx
+      | strong
+      | sub
+      | sup
+      | tt
+      | xref)*
+    }
+  refcontent =
+    element refcontent {
+      attribute xml:base { text }?,
+      attribute xml:lang { text }?,
+      (text | bcp14 | em | strong | sub | sup | tt)*
+    }
+  start = rfc

```

## Index

### A

- abbrev attribute
  - in organization element 40
  - in title element 69
- abstract element 11, 97
  - anchor attribute 11
  - inside front 33
- address element 11, 97
  - inside author 17
- align attribute
  - in artwork element 14
  - in figure element 31
  - in td element 66
  - in texttable element 79
  - in th element 68
  - in ttd element 81
- alt attribute
  - in artwork element 14
  - in figure element 31
- anchor attribute
  - in abstract element 11
  - in artwork element 15
  - in aside element 17
  - in blockquote element 20
  - in cref element 23

- in dd element 25
- in dl element 27
- in dt element 28
- in figure element 31
- in li element 36
- in ol element 39
- in reference element 43
- in referencegroup element 44
- in references element 45
- in section element 54
- in sourcecode element 58
- in t element 63
- in table element 64
- in tbody element 64
- in td element 66
- in texttable element 80
- in tfoot element 67
- in th element 68
- in thead element 69
- in tr element 70
- in ul element 71
- annotation element 12, 97
  - inside reference 43
- application/rfc+xml Media Type 82
- area element 12, 97
  - inside front 33
- artwork element 13, 97
  - align attribute 14
  - alt attribute 14
  - anchor attribute 15
  - height attribute 15
  - inside aside 16
  - inside blockquote 19
  - inside dd 24
  - inside figure 31
  - inside li 35
  - inside section 53
  - inside td 65
  - inside th 67
  - name attribute 15
  - src attribute 15
  - type attribute 15
  - width attribute 16
  - xml:space attribute 16
- ascii attribute
  - in city element 21
  - in code element 21
  - in country element 22

- in email element 29
- in organization element 41
- in postalLine element 42
- in region element 45
- in street element 59
- in title element 69
- asciiFullname attribute
  - in author element 17
- asciiInitials attribute
  - in author element 17
- asciiName attribute
  - in seriesInfo element 55
- asciiSurname attribute
  - in author element 18
- asciiValue attribute
  - in seriesInfo element 56
- aside element 16, 97
  - anchor attribute 17
  - inside section 53
- Attributes
  - abbrev 40, 69
  - align 14, 31, 66, 68, 79, 81
  - alt 14, 31
  - anchor 11, 15, 17, 20, 23, 25, 27-28, 31, 36, 39, 43-45, 54, 58, 63-64, 66-71, 80
  - ascii 21-22, 29, 41-42, 45, 59, 69
  - asciiFullname 17
  - asciiInitials 17
  - asciiName 55
  - asciiSurname 18
  - asciiValue 56
  - blankLines 81
  - border 66, 68, 70
  - category 49
  - cite 20
  - colspan 66, 69
  - consensus 49
  - counter 77
  - day 24
  - display 23
  - displayFormat 46
  - docName 50
  - empty 71
  - format 73
  - fullname 18
  - group 39
  - hangIndent 77
  - hanging 27

hangText 63  
height 15, 32  
href 36  
indexInclude 50  
initials 18  
ipr 50  
iprExtract 51  
item 34  
keepWithNext 63  
keepWithPrevious 63  
month 24  
name 15, 56, 58  
number 51  
numbered 54  
obsoletes 51  
octets 76  
pageno 74  
prepTime 51  
primary 34  
quotedFrom 20  
quoteTitle 43  
rel 36  
relative 48  
removeInRFC 38, 54  
role 18  
rowspan 66, 69  
section 49  
seriesNo 51  
sortRefs 51  
source 23  
spacing 27, 39, 72  
src 15, 32, 58  
start 39  
style 77, 79-80  
subitem 34  
submissionType 52  
suppress-title 32, 80  
surname 18  
symRefs 52  
target 26, 30, 44, 49, 74, 76  
TheRealDefinition 75  
title 32, 38, 45, 54, 80  
to 26  
toc 55  
tocDepth 52  
tocInclude 52  
type 15, 39, 58, 76  
updates 53

- value 56
- version 53
- width 16, 32, 81
- xml:space 16, 79
- year 24
- author element 17, 97
  - asciiFullname attribute 17
  - asciiInitials attribute 17
  - asciiSurname attribute 18
  - fullname attribute 18
  - initials attribute 18
  - inside front 33
  - role attribute 18
  - surname attribute 18

## B

- back element 18, 97
  - inside rfc 49
- bcpl4 element 19, 97
  - inside annotation 12
  - inside blockquote 20
  - inside c 75
  - inside dd 25
  - inside dt 27
  - inside em 28
  - inside li 35
  - inside postamble 77
  - inside preamble 78
  - inside refcontent 42
  - inside strong 60
  - inside sub 61
  - inside sup 61
  - inside t 62
  - inside td 65
  - inside th 67
  - inside tt 70
- blankLines attribute
  - in vspace element 81
- blockquote element 19, 97
  - anchor attribute 20
  - cite attribute 20
  - inside section 53
  - quotedFrom attribute 20
- boilerplate element 21, 97
  - inside front 33
- border attribute
  - in td element 66
  - in th element 68

- in tr element 70
- br element 21, 97
  - inside td 65
  - inside th 67

## C

- c element 75, 97
  - inside texttable 79
- category attribute
  - in rfc element 49
- cite attribute
  - in blockquote element 20
- city element 21, 97
  - ascii attribute 21
  - inside postal 41
- code element 21, 97
  - ascii attribute 21
  - inside postal 41
- colspan attribute
  - in td element 66
  - in th element 69
- consensus attribute
  - in rfc element 49
- counter attribute
  - in list element 77
- country element 22, 97
  - ascii attribute 22
  - inside postal 41
- cref element 22, 97
  - anchor attribute 23
  - display attribute 23
  - inside annotation 12
  - inside blockquote 20
  - inside c 75
  - inside dd 25
  - inside dt 28
  - inside em 29
  - inside li 35
  - inside name 37
  - inside postamble 77
  - inside preamble 78
  - inside strong 60
  - inside sub 61
  - inside sup 61
  - inside t 62
  - inside td 65
  - inside th 67
  - inside tt 71

inside ttrcol 80  
source attribute 23

## D

date element 23, 97  
  day attribute 24  
  inside front 33  
  month attribute 24  
  year attribute 24  
day attribute  
  in date element 24  
dd element 24, 97  
  anchor attribute 25  
  inside dl 27  
display attribute  
  in cref element 23  
displayFormat attribute  
  in relref element 46  
displayreference element 26, 97  
  inside back 18  
  target attribute 26  
  to attribute 26  
dl element 26, 97  
  anchor attribute 27  
  hanging attribute 27  
  inside abstract 11  
  inside aside 16  
  inside blockquote 19  
  inside dd 25  
  inside li 35  
  inside note 38  
  inside section 53  
  inside td 65  
  inside th 67  
  spacing attribute 27  
docName attribute  
  in rfc element 50  
dt element 27, 97  
  anchor attribute 28  
  inside dl 27

## E

## Elements

abstract 11, 33  
address 11, 17  
annotation 12, 43  
area 12, 33  
artwork 13, 16, 19, 24, 31, 35, 53, 65, 67



aside 16, 53  
author 17, 33  
back 18, 49  
bcp14 12, 19-20, 25, 27-28, 35, 42, 60-62, 65, 67, 70, 75,  
77-78  
blockquote 19, 53  
boilerplate 21, 33  
br 21, 65, 67  
c 75, 79  
city 21, 41  
code 21, 41  
country 22, 41  
cref 12, 20, 22, 25, 28-29, 35, 37, 60-62, 65, 67, 71, 75,  
77-78, 80  
date 23, 33  
dd 24, 27  
displayreference 18, 26  
dl 11, 16, 19, 25-26, 35, 38, 53, 65, 67  
dt 27  
em 12, 20, 22, 25, 28, 35, 43, 60-62, 65, 67, 71, 75, 77-78  
email 11, 29  
eref 12, 20, 22, 25, 28-29, 35, 37, 60-62, 65, 68, 71, 75,  
77-78, 80  
facsimile 11, 76  
figure 16, 19, 25, 30, 35, 53, 65, 68  
format 43, 76  
front 32, 43, 49  
iref 12, 16, 20, 25, 28-29, 31, 33, 35, 53, 60-62, 64-65, 68,  
71, 75, 77-78, 80  
keyword 33-34  
li 34, 38, 71  
link 36, 49  
list 16, 62, 76  
middle 37, 49  
name 31, 37-38, 45, 53, 64, 79  
note 33, 37  
ol 11, 16, 19, 25, 35, 38, 53, 65, 68  
organization 17, 40  
phone 11, 41  
postal 11, 41  
postalLine 42  
postamble 31, 77, 79  
preamble 31, 78-79  
refcontent 42-43  
reference 43-45  
referencegroup 44-45  
references 18, 44  
region 41, 45

relref 12, 20, 22, 25, 28-29, 35, 37, 45, 60-62, 65, 68, 71  
rfc 49  
section 18, 21, 37, 53-54  
seriesInfo 33, 43, 55  
sourcecode 19, 25, 31, 35, 54, 57, 65, 68  
spanx 12, 62, 75, 77-78  
street 41, 59  
strong 12, 20, 22, 25, 28-29, 35, 43, 59, 61-62, 65, 68, 71,  
75, 77-78  
sub 12, 20, 22, 25, 28-29, 35, 43, 60, 62, 65, 68, 71, 76-78  
sup 12, 20, 22, 25, 28-29, 35, 43, 60-62, 66, 68, 71, 76-78  
svg 14, 75  
t 11, 16, 19, 25, 35, 38, 54, 62, 65, 67, 77  
table 17, 54, 64  
tbody 64  
td 65, 70  
texttable 54, 79  
tfoot 64, 67  
th 67, 70  
thead 64, 69  
title 33, 69  
tr 64, 67, 69-70  
tt 12, 20, 23, 25, 28-29, 35, 37, 43, 60-63, 66, 68, 70, 76,  
78  
ttcol 79-80  
ul 11, 17, 19, 25, 35, 38, 54, 66, 68, 71  
uri 11, 72  
vspace 63, 81  
workgroup 33, 72  
xref 12, 20, 23, 25, 28-29, 35, 37, 60-63, 66, 68, 71-72, 76,  
78, 80  
em element 28, 97  
  inside annotation 12  
  inside blockquote 20  
  inside c 75  
  inside cref 22  
  inside dd 25  
  inside dt 28  
  inside li 35  
  inside postamble 77  
  inside preamble 78  
  inside refcontent 43  
  inside strong 60  
  inside sub 61  
  inside sup 61  
  inside t 62  
  inside td 65  
  inside th 67

- inside tt 71
- email element 29, 97
  - ascii attribute 29
  - inside address 11
- empty attribute
  - in ul element 71
- eref element 29, 97
  - inside annotation 12
  - inside blockquote 20
  - inside c 75
  - inside cref 22
  - inside dd 25
  - inside dt 28
  - inside em 29
  - inside li 35
  - inside name 37
  - inside postamble 77
  - inside preamble 78
  - inside strong 60
  - inside sub 61
  - inside sup 61
  - inside t 62
  - inside td 65
  - inside th 68
  - inside tt 71
  - inside ttcoll 80
  - target attribute 30

## F

- facsimile element 76, 97
  - inside address 11
- figure element 30, 97
  - align attribute 31
  - alt attribute 31
  - anchor attribute 31
  - height attribute 32
  - inside aside 16
  - inside blockquote 19
  - inside dd 25
  - inside li 35
  - inside section 53
  - inside td 65
  - inside th 68
  - src attribute 32
  - suppress-title attribute 32
  - title attribute 32
  - width attribute 32
- format attribute

- in xref element 73
- format element 76, 97
  - inside reference 43
  - octets attribute 76
  - target attribute 76
  - type attribute 76
- front element 32, 97
  - inside reference 43
  - inside rfc 49
- fullname attribute
  - in author element 18

## G

- group attribute
  - in ol element 39

## H

- hangIndent attribute
  - in list element 77
- hanging attribute
  - in dl element 27
- hangText attribute
  - in t element 63
- height attribute
  - in artwork element 15
  - in figure element 32
- href attribute
  - in link element 36

## I

- indexInclude attribute
  - in rfc element 50
- initials attribute
  - in author element 18
- ipr attribute
  - '\*2026' 90
  - '\*3667' 90
  - '\*3978' 90
  - '\*trust200811' 90
  - '\*trust200902' 88
  - 'noDerivativesTrust200902' 89
  - 'noModificationTrust200902' 89
  - 'pre5378Trust200902' 89
  - 'trust200902' 89
  - in rfc element 50
- iprExtract attribute
  - in rfc element 51
- iref element 33, 97

- inside annotation 12
- inside aside 16
- inside blockquote 20
- inside c 75
- inside dd 25
- inside dt 28
- inside em 29
- inside figure 31
- inside li 35
- inside postamble 77
- inside preamble 78
- inside section 53
- inside strong 60
- inside sub 61
- inside sup 62
- inside t 62
- inside table 64
- inside td 65
- inside th 68
- inside tt 71
- inside ttdcol 80
- item attribute 34
- primary attribute 34
- subitem attribute 34
- item attribute
  - in iref element 34

## K

- keepWithNext attribute
  - in t element 63
- keepWithPrevious attribute
  - in t element 63
- keyword element 34, 97
  - inside front 33

## L

- li element 34, 97
  - anchor attribute 36
  - inside ol 38
  - inside ul 71
- link element 36, 97
  - href attribute 36
  - inside rfc 49
  - rel attribute 36
- list element 76, 97
  - counter attribute 77
  - hangIndent attribute 77
  - inside aside 16

- inside t 62
- style attribute 77

## M

- Media Type
  - application/rfc+xml 82
- middle element 37, 97
  - inside rfc 49
- month attribute
  - in date element 24

## N

- name attribute
  - in artwork element 15
  - in seriesInfo element 56
  - in sourcecode element 58
- name element 37, 97
  - inside figure 31
  - inside note 38
  - inside references 45
  - inside section 53
  - inside table 64
  - inside texttable 79
- note element 37, 97
  - inside front 33
  - removeInRFC attribute 38
  - title attribute 38
- number attribute
  - in rfc element 51
- numbered attribute
  - in section element 54

## O

- obsoletes attribute
  - in rfc element 51
- octets attribute
  - in format element 76
- ol element 38, 97
  - anchor attribute 39
  - group attribute 39
  - inside abstract 11
  - inside aside 16
  - inside blockquote 19
  - inside dd 25
  - inside li 35
  - inside note 38
  - inside section 53
  - inside td 65

- inside th 68
- spacing attribute 39
- start attribute 39
- type attribute 39
- organization element 40, 97
  - abbrev attribute 40
  - ascii attribute 41
  - inside author 17

## P

- pageno attribute
  - in xref element 74
- phone element 41, 97
  - inside address 11
- postal element 41, 97
  - inside address 11
- postalLine element 42, 97
  - ascii attribute 42
  - inside postal 42
- postamble element 77, 97
  - inside figure 31
  - inside texttable 79
- preamble element 78, 97
  - inside figure 31
  - inside texttable 79
- prepTime attribute
  - in rfc element 51
- primary attribute
  - in iref element 34

## Q

- quotedFrom attribute
  - in blockquote element 20
- quoteTitle attribute
  - in reference element 43

## R

- refcontent element 42, 97
  - inside reference 43
- reference element 43, 97
  - anchor attribute 43
  - inside referencegroup 44
  - inside references 45
  - quoteTitle attribute 43
  - target attribute 44
- referencegroup element 44, 97
  - anchor attribute 44
  - inside references 45

- references element 44, 97
  - anchor attribute 45
  - inside back 18
  - title attribute 45
- region element 45, 97
  - ascii attribute 45
  - inside postal 41
- rel attribute
  - in link element 36
- relative attribute
  - in relref element 48
- relref element 45, 97
  - displayFormat attribute 46
  - inside annotation 12
  - inside blockquote 20
  - inside cref 22
  - inside dd 25
  - inside dt 28
  - inside em 29
  - inside li 35
  - inside name 37
  - inside strong 60
  - inside sub 61
  - inside sup 62
  - inside t 62
  - inside td 65
  - inside th 68
  - inside tt 71
  - relative attribute 48
  - section attribute 49
  - target attribute 49
- removeInRFC attribute
  - in note element 38
  - in section element 54
- rfc element 49, 97
  - category attribute 49
  - consensus attribute 49
  - docName attribute 50
  - indexInclude attribute 50
  - ipr attribute 50
  - iprExtract attribute 51
  - number attribute 51
  - obsoletes attribute 51
  - prepTime attribute 51
  - seriesNo attribute 51
  - sortRefs attribute 51
  - submissionType attribute 52
  - symRefs attribute 52



- tocDepth attribute 52
- tocInclude attribute 52
- updates attribute 53
- version attribute 53
- role attribute
  - in author element 18
- rowspan attribute
  - in td element 66
  - in th element 69

## S

- section attribute
  - in relref element 49
- section element 53, 97
  - anchor attribute 54
  - inside back 18
  - inside boilerplate 21
  - inside middle 37
  - inside section 54
  - numbered attribute 54
  - removeInRFC attribute 54
  - title attribute 54
  - toc attribute 55
- seriesInfo element 55, 97
  - asciiName attribute 55
  - asciiValue attribute 56
  - inside front 33
  - inside reference 43
  - name attribute 56
  - value attribute 56
- seriesNo attribute
  - in rfc element 51
- sortRefs attribute
  - in rfc element 51
- source attribute
  - in cref element 23
- sourcecode element 57, 97
  - anchor attribute 58
  - inside blockquote 19
  - inside dd 25
  - inside figure 31
  - inside li 35
  - inside section 54
  - inside td 65
  - inside th 68
  - name attribute 58
  - src attribute 58
  - type attribute 58

- spacing attribute
  - in dl element 27
  - in ol element 39
  - in ul element 72
- spanx element 78, 97
  - inside annotation 12
  - inside c 75
  - inside postamble 77
  - inside preamble 78
  - inside t 62
  - style attribute 79
  - xml:space attribute 79
- src attribute
  - in artwork element 15
  - in figure element 32
  - in sourcecode element 58
- start attribute
  - in ol element 39
- street element 59, 97
  - ascii attribute 59
  - inside postal 41
- strong element 59, 97
  - inside annotation 12
  - inside blockquote 20
  - inside c 75
  - inside cref 22
  - inside dd 25
  - inside dt 28
  - inside em 29
  - inside li 35
  - inside postamble 77
  - inside preamble 78
  - inside refcontent 43
  - inside sub 61
  - inside sup 62
  - inside t 62
  - inside td 65
  - inside th 68
  - inside tt 71
- style attribute
  - in list element 77
  - in spanx element 79
  - in texttable element 80
- sub element 60, 97
  - inside annotation 12
  - inside blockquote 20
  - inside c 76
  - inside cref 22

- inside dd 25
- inside dt 28
- inside em 29
- inside li 35
- inside postamble 77
- inside preamble 78
- inside refcontent 43
- inside strong 60
- inside t 62
- inside td 65
- inside th 68
- inside tt 71
- subitem attribute
  - in irref element 34
- submissionType attribute
  - in rfc element 52
- sup element 61, 97
  - inside annotation 12
  - inside blockquote 20
  - inside c 76
  - inside cref 22
  - inside dd 25
  - inside dt 28
  - inside em 29
  - inside li 35
  - inside postamble 77
  - inside preamble 78
  - inside refcontent 43
  - inside strong 60
  - inside t 62
  - inside td 66
  - inside th 68
  - inside tt 71
- suppress-title attribute
  - in figure element 32
  - in texttable element 80
- surname attribute
  - in author element 18
- svg element 75, 97
  - inside artwork 14
  - TheRealDefinition attribute 75
- symRefs attribute
  - in rfc element 52

## T

- t element 62, 97
  - anchor attribute 63
  - hangText attribute 63

- inside abstract 11
- inside aside 16
- inside blockquote 19
- inside dd 25
- inside li 35
- inside list 77
- inside note 38
- inside section 54
- inside td 65
- inside th 67
- keepWithNext attribute 63
- keepWithPrevious attribute 63
- table element 64, 97
  - anchor attribute 64
  - inside aside 17
  - inside section 54
- target attribute
  - in displayreference element 26
  - in eref element 30
  - in format element 76
  - in reference element 44
  - in relref element 49
  - in xref element 74
- tbody element 64, 97
  - anchor attribute 64
  - inside table 64
- td element 65, 97
  - align attribute 66
  - anchor attribute 66
  - border attribute 66
  - colspan attribute 66
  - inside tr 70
  - rowspan attribute 66
- texttable element 79, 97
  - align attribute 79
  - anchor attribute 80
  - inside section 54
  - style attribute 80
  - suppress-title attribute 80
  - title attribute 80
- tfoot element 67, 97
  - anchor attribute 67
  - inside table 64
- th element 67, 97
  - align attribute 68
  - anchor attribute 68
  - border attribute 68
  - colspan attribute 69

- inside tr 70
- rowspan attribute 69
- thead element 69, 97
  - anchor attribute 69
  - inside table 64
- TheRealDefinition attribute
  - in svg element 75
- title attribute
  - in figure element 32
  - in note element 38
  - in references element 45
  - in section element 54
  - in texttable element 80
- title element 69, 97
  - abbrev attribute 69
  - ascii attribute 69
  - inside front 33
- to attribute
  - in displayreference element 26
- toc attribute
  - in section element 55
- tocDepth attribute
  - in rfc element 52
- tocInclude attribute
  - in rfc element 52
- tr element 70, 97
  - anchor attribute 70
  - border attribute 70
  - inside tbody 64
  - inside tfoot 67
  - inside thead 69
- tt element 70, 97
  - inside annotation 12
  - inside blockquote 20
  - inside c 76
  - inside cref 23
  - inside dd 25
  - inside dt 28
  - inside em 29
  - inside li 35
  - inside name 37
  - inside postamble 78
  - inside preamble 78
  - inside refcontent 43
  - inside strong 60
  - inside sub 61
  - inside sup 62
  - inside t 63

- inside td 66
- inside th 68
- ttcol element 80, 97
  - align attribute 81
  - inside texttable 79
  - width attribute 81
- type attribute
  - in artwork element 15
  - in format element 76
  - in ol element 39
  - in sourcecode element 58

## U

- ul element 71, 97
  - anchor attribute 71
  - empty attribute 71
  - inside abstract 11
  - inside aside 17
  - inside blockquote 19
  - inside dd 25
  - inside li 35
  - inside note 38
  - inside section 54
  - inside td 66
  - inside th 68
  - spacing attribute 72
- updates attribute
  - in rfc element 53
- uri element 72, 97
  - inside address 11

## V

- value attribute
  - in seriesInfo element 56
- version attribute
  - in rfc element 53
- vspace element 81, 97
  - blankLines attribute 81
  - inside t 63

## W

- width attribute
  - in artwork element 16
  - in figure element 32
  - in ttcol element 81
- workgroup element 72, 97
  - inside front 33

## X

- xml:space attribute
  - in artwork element 16
  - in spanx element 79
- xref element 72, 97
  - format attribute 73
  - inside annotation 12
  - inside blockquote 20
  - inside c 76
  - inside cref 23
  - inside dd 25
  - inside dt 28
  - inside em 29
  - inside li 35
  - inside name 37
  - inside postamble 78
  - inside preamble 78
  - inside strong 60
  - inside sub 61
  - inside sup 62
  - inside t 63
  - inside td 66
  - inside th 68
  - inside tt 71
  - inside ttcoll 80
  - pageno attribute 74
  - target attribute 74
- xref formats
  - counter 73
  - default 73
  - none 74
  - title 74

## Y

- year attribute
  - in date element 24

## Author's Address

Paul Hoffman  
ICANN

EMail: paul.hoffman@icann.org

