The ’XML2RFC’ version 2 Vocabulary
draft-reschke-xml2rfc-07

Abstract

This document defines the ’XML2RFC’ version 2 vocabulary; an XML-based language used for writing RFCs and Internet-Drafts.

Editorial Note (To be removed by RFC Editor)

Discussion of this draft takes place on the XML2RFC mailing list (xml2rfc@ietf.org), which has its home page at <https://www.ietf.org/mailman/listinfo/xml2rfc>.

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 October 10, 2014.

Copyright Notice

Copyright (c) 2014 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 ................................................. 4
   1.1. Syntax Notation ....................................... 4
2. Elements ..................................................... 4
   2.1. <abstract> ........................................... 4
   2.2. <address> ........................................... 4
   2.3. <annotation> ......................................... 5
   2.4. <area> ............................................... 6
   2.5. <artwork> ............................................. 6
   2.6. <author> ............................................. 8
   2.7. <back> ................................................ 9
   2.8. <c> .................................................. 10
   2.9. <city> ............................................... 10
   2.10. <code> .............................................. 10
   2.11. <country> .......................................... 10
   2.12. <cref> ............................................... 11
   2.13. <date> ............................................... 11
   2.14. <email> .............................................. 12
   2.15. <eref> ............................................... 12
   2.16. <facsimile> ......................................... 13
   2.17. <figure> ............................................ 13
   2.18. <format> ............................................. 15
   2.19. <front> .............................................. 15
   2.20. <iref> ............................................... 16
   2.21. <keyword> ........................................... 17
   2.22. <list> ................................................ 17
   2.23. <middle> ............................................ 19
   2.24. <note> ................................................ 19
   2.25. <organization> ....................................... 20
   2.26. <phone> .............................................. 20
   2.27. <postal> ............................................ 20
   2.28. <postamble> ......................................... 21
   2.29. <preamble> .......................................... 21
   2.30. <reference> .......................................... 22
   2.31. <references> ........................................ 22
   2.32. <region> ............................................. 23
   2.33. <rfc> ................................................ 23
   2.34. <section> ............................................ 26
   2.35. <seriesInfo> ......................................... 27
   2.36. <spanx> .............................................. 28
   2.37. <street> ............................................. 29
   2.38. <t> .................................................. 29
1. Introduction

This document describes version 2 ('v2') of the 'XML2RFC' vocabulary; an XML-based language ('Extensible Markup Language', [XML]) used for writing RFCs ([RFCSTYLE]) and Internet-Drafts ([IDGUIDE]).

It obsoletes the original version ("v1") [RFC2629], which contained the original language definition, and which was subsequently extended ("v2"). Furthermore, it discusses potential extensions in a future revision ("v3").

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

1.1. 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.

2.1. <abstract>

Contains the abstract of the document. The abstract ought to be self-contained and thus should not contain references or unexpanded abbreviations. See Section 4.3 of [RFCSTYLE] for more information.

This element appears as child element of: <front> (Section 2.19).

Content model:

One or more <t> elements (Section 2.38)

2.2. <address>

Provides address information for the author.
This element appears as child element of: `<author>` (*Section 2.6*).

Content model:

In this order:

1. One optional `<postal>` element (*Section 2.27*)
2. One optional `<phone>` element (*Section 2.26*)
3. One optional `<facsimile>` element (*Section 2.16*)
4. One optional `<email>` element (*Section 2.14*)
5. One optional `<uri>` element (*Section 2.42*)

### 2.3. `<annotation>`

Provides additional prose augmenting a bibliographical reference.

For instance:

```xml
<annotation>
  Latest version available at <eref target='http://www.w3.org/TR/xml'/>.
</annotation>
```

...will generate the text used in the reference for [XML].

This element appears as child element of: `<reference>` (*Section 2.30*).

Content model:

In any order:

- Text
- `<xref>` elements (*Section 2.45*)
- `<eref>` elements (*Section 2.15*)
- `<iref>` elements (*Section 2.20*)
- `<cref>` elements (*Section 2.12*)
- `<spanx>` elements (*Section 2.36*)
2.4. <area>

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

The value ought to be either the fullname or the abbreviation of one of the IETF areas as listed on <http://www.ietf.org/iesg/area.html>: "Applications", "app", "General", "gen", "Internet", "int", "Operations and Management", "ops", "Real-time Applications and Infrastructure", "rai", "Routing", "rtg", "Security", "sec", "Transport", "tsv".

This element appears as child element of: <front> (Section 2.19).

Content model: only text content.

2.5. <artwork>

This element allows the inclusion of "artwork" into the document.

<artwork> is the only element in the vocabulary that provides full control of horizontal whitespace and line breaks, and thus is used for a variety of things, such as:

- diagrams ("line art"),
- source code,
- formal languages (such as ABNF or the RNC notation used in this document),
- complex tables, or
- protocol unit diagrams.

Alternatively, the "src" attribute allows referencing an external graphics file, such as a bitmap or a vector drawing. In this case, the textual content acts as fallback for output formats 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. Note that RFCs occasionally are published with enhanced diagrams; a recent example is [RFC5598].

This element appears as child element of: <figure> (Section 2.17).

Content model:

Text
2.5.1. 'align' attribute

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

Allowed values:

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

2.5.2. 'alt' attribute

Alternative text description of the artwork (not just the caption).

2.5.3. 'height' attribute

The suggested height of the graphics included using the "src" attribute.

This attribute is format-dependent and ought to be avoided.

When generating HTML output, current implementations copy the attribute "as is". For other output formats it is usually ignored.

2.5.4. 'name' attribute

A filename suitable for the contents (such as for extraction to a local file).

This attribute generally isn’t used for document generation, but it can be helpful for other kinds of tools (such as automated syntax checkers which work by extracting the source code).

2.5.5. 'src' attribute

The URI of a graphics file.

Note that this can be a "data" URI ([RFC2397]) as well, in which case the graphics file essentially is in-lined.

2.5.6. 'type' attribute

Specifies the type of the artwork.

The value either is a well-known keyword (such as "abnf"), or an
Internet Media Type (see [RFC2046]).

How it is used depends on context and application. For instance, a formatter can attempt to syntax-highlight code in certain known languages.

2.5.7. ‘width’ attribute

The suggested width of the graphics included using the "src" attribute.

This attribute is format-dependent and ought to be avoided.

When generating HTML output, current implementations copy the attribute "as is". For other output formats it is usually ignored.

2.5.8. ‘xml:space’ attribute

Determines whitespace handling.

"preserve" is both the default value and the only meaningful setting anyway (because that’s what the <artwork> element is for).

See also Section 2.10 of [XML].

Allowed values:

- "default"
- "preserve" (default)

2.6. <author>

Provides information about a document author.

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 Section 4.12 of [RFCSTYLE]).

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 child element of: <front> (Section 2.19).

Content model:

In this order:

1. One optional <organization> element (Section 2.25)
2. One optional <address> element (Section 2.2)

2.6.1. 'fullname' attribute

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

2.6.2. 'initials' attribute

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

Initials should be provided as a whitespace separated list of pairs of a letter and a dot.

2.6.3. 'role' attribute

Specifies the role the author had in creating the document.

Allowed values:
- "editor"

2.6.4. 'surname' attribute

The author’s surname.

2.7. <back>

Contains the "back" part of the document: the references and appendices.

This element appears as child element of: <rfc> (Section 2.33).

Content model:

In this order:
1. Optional <references> elements (Section 2.31)

2. Optional <section> elements (Section 2.34)

2.8. <c>

Provides the content of a cell in a table.

This element appears as child element of: <texttable> (Section 2.39).

Content model:

In any order:

- Text
- <xref> elements (Section 2.45)
- <eref> elements (Section 2.15)
- <iref> elements (Section 2.20)
- <cref> elements (Section 2.12)
- <spanx> elements (Section 2.36)

2.9. <city>

Gives the city name in a postal address.

This element appears as child element of: <postal> (Section 2.27).

Content model: only text content.

2.10. <code>

Gives the postal region code.

This element appears as child element of: <postal> (Section 2.27).

Content model: only text content.

2.11. <country>

Gives the country in a postal address.

This element appears as child element of: <postal> (Section 2.27).
2.12.  <cref>

Represents a comment.

Comments can be used in a document while it is work-in-progress. They usually appear either inline and visually highlighted, at the end of the document (depending on file format and settings of the formatter), or not at all (when generating an RFC).

This element appears as child element of: <annotation> (Section 2.3), <c> (Section 2.8), <postamble> (Section 2.28), <preamble> (Section 2.29), and <t> (Section 2.38).

Content model: only text content.

2.12.1.  'anchor' attribute

Document-wide unique identifier for this comment. The processor will auto-generate an identifier when none is given.

The value needs to be a valid XML "Name" (Section 2.3 of [XML]).

2.12.2.  'source' attribute

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

2.13.  <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.

In the first case, it defines the publication date, which, when producing Internet-Drafts, will be used for computing the expiration date (see Section 8 of [IDGUIDE]). When "year", "month" or "day" are left out, the processor will attempt to use the current system date if the attributes that are specified do match the system date.

Note that month names need to match the full (English) month name ("January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", or "December") in order for expiration calculations to work (some implementations might support additional formats, though).
In the second case, the date information will be embedded as-is into the reference text. Therefore, also vague dates ("ca. 2000"), date ranges, and so on, are allowed.

This element appears as child element of: <front> (Section 2.19).

Content model: this element does not have any contents.

2.13.1. 'day' attribute

Day of publication.

2.13.2. 'month' attribute

Month of publication.

2.13.3. 'year' attribute

Year of publication.

2.14. <email>

Provides an email address.

The value is expected to be the scheme-specific part of a "mailto" URI (so does not include the prefix "mailto:"). See Section 2 of [RFC6068] for details.

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

Content model: only text content.

2.15. <eref>

Represents an "external" link (as specified in the "target" attribute).

If the element has text content, that content will be used. Otherwise, the value of the target attribute will be inserted in angle brackets ([RFC3986], Appendix C).

This element appears as child element of: <annotation> (Section 2.3), <c> (Section 2.8), <postamble> (Section 2.28), <preamble> (Section 2.29), and <t> (Section 2.38).

Content model: only text content.
2.15.1. ‘target’ attribute (mandatory)

URI of the link target (see Section 3 of [RFC3986]).

2.16. <facsimile>

Represents the phone number of a fax machine.

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 child element of: <address> (Section 2.2).

Content model: only text content.

2.17. <figure>

This element is used to represent a figure, consisting of an optional
preamble, the actual figure, an optional postamble, and an optional
title.

This element appears as child element of: <section> (Section 2.34),
and <t> (Section 2.38).

Content model:

In this order:

1. Optional <iref> elements (Section 2.20)
2. One optional <preamble> element (Section 2.29)
3. One <artwork> element (Section 2.5)
4. One optional <postamble> element (Section 2.28)

2.17.1. ‘align’ attribute

Used to change the alignment of <preamble> and <postamble>.

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

Allowed values:

- "left" (default)
2.17.2. ‘alt’ attribute

Duplicates functionality available on <artwork>; avoid it.

2.17.3. ‘anchor’ attribute

Document-wide unique identifier for this figure.

Furthermore, the presence of this attribute causes the figure to be numbered.

The value needs to be a valid XML "Name" (Section 2.3 of [XML]).

2.17.4. ‘height’ attribute

Duplicates functionality available on <artwork>; avoid it.

2.17.5. ‘src’ attribute

Duplicates functionality available on <artwork>; avoid it.

2.17.6. ‘suppress-title’ attribute

Figures that have an "anchor" attribute will automatically get an autogenerated title (such as "Figure 1"), even if the "title" attribute is absent. Setting this attribute to "true" will prevent this.

Allowed values:

- "true"
- "false" (default)

2.17.7. ‘title’ attribute

The title for the figure; this usually appears on a line after the figure.

2.17.8. ‘width’ attribute

Duplicates functionality available on <artwork>; avoid it.
2.18. <format>

Provides a link to an additional format variant for a reference.

Note that these additional links are neither used in published RFCs, nor supported by all tools. If the goal is to provide a single URI for a reference, the "target" attribute on <reference> can be used instead.

This element appears as child element of: <reference> (Section 2.30).

Content model: this element does not have any contents.

2.18.1. 'octets' attribute

Octet length of linked-to document.

2.18.2. 'target' attribute

URI of document.

2.18.3. 'type' attribute (mandatory)

The type of the linked-to document, such as "TXT", "HTML", or "PDF".

2.19. <front>

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

This element appears as child element of: <reference> (Section 2.30), and <rfc> (Section 2.33).

Content model:

In this order:

1. One <title> element (Section 2.40)
2. One or more <author> elements (Section 2.6)
3. One <date> element (Section 2.13)
4. Optional <area> elements (Section 2.4)
5. Optional <workgroup> elements (Section 2.44)
6. Optional `<keyword>` elements (Section 2.21)

7. One optional `<abstract>` element (Section 2.1)

8. Optional `<note>` elements (Section 2.24)

2.20. `<iref>`

Provides terms for the document’s index.

Index entries can be either single items (when just the "item" attribute is given) or nested items (by specifying "subitem" as well).

For instance:

```xml
<iref item="Grammar" subitem="item"/>
```

will produce an index entry for "Grammar, item".

This element appears as child element of: `<annotation>` (Section 2.3), `<c>` (Section 2.8), `<figure>` (Section 2.17), `<postamble>` (Section 2.28), `<preamble>` (Section 2.29), `<section>` (Section 2.34), and `<t>` (Section 2.38).

Content model: this element does not have any contents.

2.20.1. 'item' attribute (mandatory)

The item to include.

2.20.2. 'primary' attribute

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

Allowed values:

- "true"
- "false" (default)

2.20.3. 'subitem' attribute

The subitem to include.
2.21.  <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 formats.

This element appears as child element of: <front> (Section 2.19).

Content model: only text content.

2.22.  <list>

Delineates a text list.

Each list item is represented by a <t> element. The vocabulary currently does not directly support list items consisting of multiple paragraphs; if this is needed, <vspace> (Section 2.43) can be used as workaround.

This element appears as child element of: <t> (Section 2.38).

Content model:

One or more <t> elements (Section 2.38)

2.22.1.  ‘counter’ attribute

This attribute holds a token that serves as an identifier for a counter. The intended use is continuation of lists.

Note that this attribute functions only when the style attribute is using the "format..." syntax (Section 2.22.3); otherwise, it is ignored.

2.22.2.  ‘hangIndent’ attribute

For list styles with potentially wide labels, this attribute can override the default indentation level, measured in characters.

Note that it only affects style with variable-width labels ("format..." and "hanging", see below), and it may not affect formats in which the list item text appears _below_ the label.
2.22.3. ‘style’ attribute

This attribute is used to control the display of a list.

The value of this attribute is inherited by any nested lists that do not have this attribute set. It may be set to:

"empty" (or not set)

For unlabeled list items; it can also be used for indentation purposes (this is the default value).

"hanging"

For lists where the items are labeled with a piece of text.

The label text is specified in the ‘hangText’ attribute of the <t> element (Section 2.38.2).

"letters"

For ordered lists using letters as labels (lowercase letters followed by a period; after "z", it rolls over to a two-letter format). For nested lists, processors usually flip between uppercase and lowercase.

"numbers"

For ordered lists using numbers as labels.

"symbols"

For unordered (bulleted) lists.

The style of the bullets is chosen automatically be the processor (some implementations allow overriding the default using a processing instruction).

And, finally:

"format ..."

For lists with customized labels, consisting of fixed text and an item counter in various formats.

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, etc.)
%C uppercase letters (A, B, C, etc.)
%d decimal numbers (1, 2, 3, etc.)
%i lowercase Roman numerals (i, ii, iii, etc.)
%I uppercase Roman numerals (I, II, III, etc.)
%% represents a percent sign

Other formats are reserved for future use.

2.23. <middle>

Represents the main content of the document.

This element appears as child element of: <rfc> (Section 2.33).

Content model:

One or more <section> elements (Section 2.34)

2.24. <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 child element of: <front> (Section 2.19).

Content model:

One or more <t> elements (Section 2.38)

2.24.1. ‘title’ attribute (mandatory)

The title of the note.
2.25. <organization>

Specifies the affiliation of an author.

This information appears in both the "Author’s Address" section and on the front page ([RFCSTYLE], Section 4.1.1). If the value is long, an abbreviated variant can be specified in the "abbrev" attribute.

This element appears as child element of: <author> (Section 2.6).

Content model: only text content.

2.25.1. 'abbrev' attribute

Abbreviated variant.

2.26. <phone>

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 child element of: <address> (Section 2.2).

Content model: only text content.

2.27. <postal>

Contains child elements providing postal information.

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

Content model:

In this order:

1. One or more <street> elements (Section 2.37)

2. In any order:

   * <city> elements (Section 2.9)

   * <region> elements (Section 2.32)

   * <code> elements (Section 2.10)
* `<country>` elements ([Section 2.11](#))

### 2.28. `<postamble>`

Gives text that appears at the bottom of a figure or table.

This element appears as child element of: `<figure>` ([Section 2.17](#)), and `<texttable>` ([Section 2.39](#)).

Content model:

In any order:

- Text
- `<xref>` elements ([Section 2.45](#))
- `<eref>` elements ([Section 2.15](#))
- `<iref>` elements ([Section 2.20](#))
- `<cref>` elements ([Section 2.12](#))
- `<spanx>` elements ([Section 2.36](#))

### 2.29. `<preamble>`

Gives text that appears at the top of a figure or table.

This element appears as child element of: `<figure>` ([Section 2.17](#)), and `<texttable>` ([Section 2.39](#)).

Content model:

In any order:

- Text
- `<xref>` elements ([Section 2.45](#))
- `<eref>` elements ([Section 2.15](#))
- `<iref>` elements ([Section 2.20](#))
- `<cref>` elements ([Section 2.12](#))
- `<spanx>` elements ([Section 2.36](#))
2.30. <reference>

Represents a bibliographical reference.

This element appears as child element of: <references> (Section 2.31).

Content model:

In this order:

1. One <front> element (Section 2.19)
2. Optional <seriesInfo> elements (Section 2.35)
3. Optional <format> elements (Section 2.18)
4. Optional <annotation> elements (Section 2.3)

2.30.1. ‘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.

The value needs to be a valid XML "Name" (Section 2.3 of [XML]).

2.30.2. ‘target’ attribute

Holds the URI for the reference.

Note that depending on the <seriesInfo> element, a URI might not be needed, nor desirable, as it can be automatically generated (for instance, for RFCs).

2.31. <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"; see Section 4.8.6 of [RFCSTYLE]). This vocabulary supports the split with the "title" attribute.

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

Content model:
One or more <reference> elements (Section 2.30)

2.31. ‘title’ attribute

Provides the title for the References section (defaulting to "References").

In general, the title should be either "Normative References" or "Informative References".

2.32. <region>

Provides the region name in a postal address.

This element appears as child element of: <postal> (Section 2.27).

Content model: only text content.

2.33. <rfc>

This is the root element of the xml2rfc vocabulary.

Processors distinguish between RFC mode ("number" attribute being present) and Internet-Draft mode ("docName" attribute being present): it is invalid to specify both. Setting neither "number" nor "docName" can be useful for producing other types of document but is out-of-scope for this specification.

Content model:

In this order:

1. One <front> element (Section 2.19)
2. One <middle> element (Section 2.23)
3. One optional <back> element (Section 2.7)

2.33.1. ‘category’ attribute

Document category (see Appendix A.1).

Allowed values:

- "std"
- "bcp"
2.33.2. ‘consensus’ attribute

Affects the generated boilerplate.

See [RFC5741] for more information.

Allowed values:
- "no"
- "yes"

2.33.3. ‘docName’ attribute

For Internet-Drafts, this specifies the draft name (which appears
below the title).

Note that the file extension is not part of the draft, so in general
it should end with the current draft number ("-", plus two digits).

Furthermore, it is good practice to disambiguate current editor
copies from submitted drafts (for instance, by replacing the draft
number with the string "latest").

See Section 7 of [IDGUIDE] for further information.

2.33.4. ‘ipr’ attribute

Represents the Intellectual Property status of the document. See
Appendix A.2 for details.

Allowed values:
- "full2026"
- "noDerivativeWorks2026"
- "none"
- "full3667"
2.33.5. ‘iprExtract’ attribute

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

2.33.6. ‘number’ attribute

The number of the RFC to be produced.

2.33.7. ‘obsoletes’ attribute

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

Processors ought to parse the attribute value, so that incorrect references can be detected and, depending on output format, hyperlinks can be generated. Also, the value ought to be reformatted to insert whitespace after each comma if not already present.

2.33.8. ‘seriesNo’ attribute

When producing a document within document series (such as "STD"): the number within that series.
2.33.9. 'submissionType' attribute

The document stream.

See Section 2 of [RFC5741] for details.

Allowed values:

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

2.33.10. 'updates' attribute

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

Processors ought to parse the attribute value, so that incorrect references can be detected and, depending on output format, hyperlinks can be generated. Also, the value ought to be reformatted to insert whitespace after each comma if not already present.

2.33.11. 'xml:lang' attribute

The natural language used in the document (defaults to "en").

See Section 2.12 of [XML] for more information.

2.34. <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.

This element appears as child element of: <back> (Section 2.7), <middle> (Section 2.23), and <section> (Section 2.34).

Content model:

In this order:

1. In any order:
*  <t> elements (Section 2.38)
*  <figure> elements (Section 2.17)
*  <texttable> elements (Section 2.39)
*  <iref> elements (Section 2.20)

2. Optional <section> elements (Section 2.34)

2.34.1. 'anchor' attribute

Document-wide unique identifier for this section.

The value needs to be a valid XML "Name" (Section 2.3 of [XML]).

2.34.2. 'title' attribute (mandatory)

The title of the section.

2.34.3. 'toc' attribute

Determines whether the section is included in the Table Of Contents.

The processor usually has defaults for whether a Table Of Contents will be produced at all, and sections of which maximal depth will be included (frequently: 3). "include" and "exclude" allow overriding the processor’s default behavior for the element they are specified on (they do not affect nested elements).

Allowed values:

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

2.35. <seriesInfo>

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

This element appears as child element of: <reference> (Section 2.30).

Content model: this element does not have any contents.
2.35.1. ‘name’ attribute (mandatory)

The name of the series.

The following names trigger specific processing (such as for auto-generating links, and adding descriptions such as "work in progress"): "BCP", "FYI", "Internet-Draft", "RFC", and "STD".

2.35.2. ‘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).

2.36. <spanx>

Wraps a piece of text, indicating special formatting styles.

When generating plain text, processors usually emulate font changes using characters such as "*" and "_".

The following styles are defined:

emph Simple emphasis (this is the default).

strong Strong emphasis.

verb "Verbatim" text (usually displayed using a monospaced font face).

This element appears as child element of: <annotation> (Section 2.3), <c> (Section 2.8), <postamble> (Section 2.28), <preamble> (Section 2.29), and <t> (Section 2.38).

Content model: only text content.

2.36.1. ‘style’ attribute

The style to be used (defaults to "emph").

2.36.2. ‘xml:space’ attribute

Determines whitespace handling.

According to the DTD, the default value is "preserve". Tests however show that it doesn’t have any effect on processing; thus this
attribute will be removed in future versions of the vocabulary.

See also Section 2.10 of [XML].

 Allowed values:
   o "default"
   o "preserve" (default)

2.37. <street>

Provides a street address.

This element appears as child element of: <postal> (Section 2.27).

Content model: only text content.

2.38. <t>

Contains a paragraph of text.

This element appears as child element of: <abstract> (Section 2.1), <list> (Section 2.22), <note> (Section 2.24), and <section> (Section 2.34).

Content model:

In any order:

  o Text
  o <list> elements (Section 2.22)
  o <figure> elements (Section 2.17)
  o <xref> elements (Section 2.45)
  o <eref> elements (Section 2.15)
  o <iref> elements (Section 2.20)
  o <cref> elements (Section 2.12)
  o <spanx> elements (Section 2.36)
  o <vspace> elements (Section 2.43)
2.38.1. ‘anchor’ attribute

Document-wide unique identifier for this paragraph.

The value needs to be a valid XML "Name" (Section 2.3 of [XML]).

2.38.2. ‘hangText’ attribute

Holds the label ("hanging text") for items in lists using the "hanging" style (see Section 2.22.3).

2.39. <texttable>

Contains a table, consisting of an optional preamble, a header line, rows, an optional postamble, and an optional title.

The number of columns in the table is determined by the number of <ttcol> elements. The number of rows in the table is determined by the number of <c> elements divided by the number of columns. There is no requirement that the number of <c> elements be evenly divisible by the number of columns.

This element appears as child element of: <section> (Section 2.34).

Content model:

In this order:

1. One optional <preamble> element (Section 2.29)
2. One or more <ttcol> elements (Section 2.41)
3. Optional <c> elements (Section 2.8)
4. One optional <postamble> element (Section 2.28)

2.39.1. ‘align’ attribute

Determines the horizontal alignment of the table.

Allowed values:

- "left"
- "center" (default)
- "right"
2.39.2. 'anchor' attribute

Document-wide unique identifier for this table.

Furthermore, the presence of this attribute causes the table to be numbered.

The value needs to be a valid XML "Name" (Section 2.3 of [XML]).

2.39.3. 'style' attribute

Selects which borders should be drawn, where

- "all" means borders around all table cells,
- "full" is like "all" except no horizontal lines between table rows (except below the column titles),
- "headers" adds just a separator between column titles and rows, and
- "none" means no borders at all.

Allowed values:

- "all"
- "none"
- "headers"
- "full" (default)

2.39.4. 'suppress-title' attribute

Tables that have an "anchor" attribute will automatically get an autogenerated title (such as "Table 1"), even if the "title" attribute is absent. Setting this attribute to "true" will prevent this.

Allowed values:

- "true"
- "false" (default)
2.39.5. 'title' attribute

The title for the table; this usually appears on a line below the table body.

2.40. <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's long (~40 characters), the "abbrev" attribute is used to specified an abbreviated variant.

This element appears as child element of: <front> (Section 2.19).

Content model: only text content.

2.40.1. 'abbrev' attribute

Specifies an abbreviated variant of the document title.

2.41. <ttcol>

Contains a column heading in a table.

This element appears as child element of: <texttable> (Section 2.39).

Content model: only text content.

2.41.1. 'align' attribute

Determines the horizontal alignment within the table column.

Allowed values:

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

2.41.2. 'width' attribute

The desired column width (as integer 0..100 followed by ")").
2.42. `<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 child element of: `<address>` (Section 2.2).

Content model: only text content.

2.43. `<vspace>`

This element can be used to force the inclusion of a single line break or multiple blank lines.

Note that this is a purely presentational element and thus its use ought to be avoided.

This element appears as child element of: `<t>` (Section 2.38).

Content model: this element does not have any contents.

2.43.1. ‘blankLines’ attribute

Number of blank lines to be inserted, where "0" indicates a single line break (defaults to "0").

For paged output formats, no additional blank lines should be generated after a page break.

2.44. `<workgroup>`

This element is used to specify the Working Group the document originates from, 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 2.33.9).

This element appears as child element of: `<front>` (Section 2.19).

Content model: only text content.
2.45. `<xref>`

Inserts a reference to a different part of a document.

The generated text depends on whether the `<xref>` is empty (in which case the processor will try to generate a meaningful text fragment), and the nature of the referenced document part.

Any element that allows the "anchor" attribute can be referenced, however there are restrictions with respect to the text content being generated. For instance, a `<t>` can be a reference target, however, because paragraphs are not (visibly) numbered, the author will have to make sure that the prose is sufficient for a reader to understand what is being referred to.

[[anchor2: This needs to be expanded with examples and with a discussion how the autogenerated text differs when `<xref>` is not empty]]

This element appears as child element of: `<annotation>` (Section 2.3), `<c>` (Section 2.8), `<postamble>` (Section 2.28), `<preamble>` (Section 2.29), and `<t>` (Section 2.38).

Content model: only text content.

2.45.1. 'format' attribute

This attribute is used to control the format of the generated reference text.

"counter"

Inserts a counter, such as the number of a section, figure, or table.

"default"

Inserts a text fragment that describes the referenced part completely, such as "Section 2", "Table 4", or "[XML]".

"none"

There will be no auto-generated text.

"title"

Inserts a title for the referenced element (usually obtained from the referenced element’s "title" attribute; some processors also
use the <title> child element or a <reference> target).

Allowed values:

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

### 2.45.2. `pageno` attribute

Unused.

It’s unclear what the purpose of this attribute is; processors seem to ignore it and it never was documented.

Allowed values:

- "true"
- "false" (default)

### 2.45.3. `target` attribute (mandatory)

Identifies the document component being referenced.

The value needs to match the value of the "anchor" attribute of another element in the document.

### 3. Special Unicode Code Points

Although the current RFC format does not allow non-ASCII Unicode characters ([UNICODE]), some of them can be used to enforce certain behaviors of formatters.

For instance:

**non-breaking space (U+00A0)**

Represents a space character where no line break should happen. This is frequently used in titles (by excluding certain space characters from the line breaking algorithm, the processor will use the remaining whitespace occurrences for line breaks).

**non-breaking hyphen (U+2011)**
Similarly, this represents a hyphen character where nevertheless
no line breaking ought to occur.

word joiner (U+2060)

Also called "zero width non-breaking space" -- can be used to
disallow line breaking between two non-whitespace characters.

Note that in order to use these characters by name, they need to be
declared either in the Document Type Definition (DTD, [XML], Section
2.9), or in the "internal subset" ([XML], Section 2.8), like that:

<?xml version="1.0"?>
<!DOCTYPE rfc [

<!-- declare nbsp and friends -->
<!ENTITY nbsp    "&#xa0;">     <!-- declare nbsp and friends -->
<!ENTITY nbhy    "&#x2011;">     <!-- declare nbsp and friends -->
<!ENTITY wj      "&#x2060;">     <!-- declare nbsp and friends -->
]>}

4. Including Files

This version of the vocabulary does not support an inclusion
mechanism on its own -- thus, a document always needs to be self-
contained.

That being said, some processors do support file inclusion using
processing instructions (Section 2.6 of [XML]).

Furthermore, XML itself allows inclusion of external content using
the "internal subset" (Section 2.8 of [XML]). Unfortunately, this
requires declaring the external data in the DTD upfront.

For instance:

<?xml version="1.0"?>
<!DOCTYPE rfc [

<!-- allow later RFC2616 reference using "&rfc2616;" -->
<!-- the data will be fetched from xml.resource.org -->
<!ENTITY rfc2616 PUBLIC
 "http://xml.resource.org/public/rfc/bibxml/reference.RFC.2616.xml">
]>}

...declares the entity "rfc2616", which then can be used in the
"references" section:

<references>
  &rfc2616;
</references>

Note that this mechanism only works for well-formed XML fragments; thus any plain text that would need to be escaped in XML can’t be included as-is.

5. Internationalization Considerations

This format is based on [XML], thus does not have any issues representing arbitrary Unicode [UNICODE] characters in text content.

However, the current canonical RFC format is restricted to US-ASCII [USASCII] characters ([RFC2223], Section 3). Future versions are likely to relax this role, and it is expected that the vocabulary will be extended so that US-ACSII alternatives can be provided when that makes sense (for instance, in contact information).

6. Security Considerations

The "name" attribute on the <artwork> element (Section 2.5.4) 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.

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

7. IANA Considerations

7.1. Internet Media Type Registration

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

This document serves as the specification for the Internet media type "application/rfc+xml". 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 [RFC3023].

Encoding considerations: Identical to those of "application/xml" as described in Section 3.2 of [RFC3023].

Security considerations: As defined in Section 6. In addition, as this media type uses the "+xml" convention, it inherits the security considerations described in Section 10 of [RFC3023].

Interoperability considerations: N/A

Published specification: This specification.

Applications that use this media type: Applications that either transform xml2rfc to output formats 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 3.2 of [RFC3023].

File extension(s): .xml

Macintosh file type code(s): TEXT

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

Intended usage: COMMON

Restrictions on usage: N/A

Author: See Authors Section.
8. Acknowledgments

Thanks to everybody who reviewed this document and provided feedback and/or specification text, in particular Brian Carpenter, Tony Hansen, Paul Hoffman, Henrik Levkowetz, Alice Russo, Tom Taylor, Jim Schaad, and Nico Williams.

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

9. References

9.1. Normative References


Latest version available at <http://www.w3.org/TR/xml>.

9.2. Informative References


[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629,
June 1999.


[TLP3.0] IETF Trust, "Legal Provisions Relating to IETF Documents", September 2009,
Appendix A. Front Page Generation

A.1. The /rfc/@category Attribute

For RFCs, the "category" determines the "maturity level" (see Section 4 of [RFC2026]). The allowed values are "std" for "Standards Track", "bcp" for "BCP", "info" for "Informational", "exp" for "Experimental", and "historic" for - surprise - "Historic".

For Internet-Drafts, the category attribute is not needed, but will appear on the front page as "Intended Status". Supplying this information can be useful to reviewers.

A.2. The /rfc/@ipr Attribute

This attribute value can take a long list of values, each of which describes an IPR policy for the document. This attribute’s values
are not the result of a grand plan, but remain simply for historic reasons. Of these values, only a few are currently in use; all others are supported by the 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 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.2.1. Current Values: ‘*trust200902’

The name for these values refers to the "IETF TRUST Legal Provisions Relating to IETF Documents", sometimes simply called the "TLP", that 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 tools will automatically produce 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.2.1.1. trust200902

This should be the default, 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.2.1.2. 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.2.1.3. 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.2.1.4. 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.2.2. Historic Values

A.2.2.1. 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>.

A.2.2.2. Historic Values: '*'3978'

The attribute values "full3978", "noModification3978" and "noDerivatives3978" are similar to their counterparts above, except that they use text specified in RFC 3978 (March 2005).

A.2.2.3. Historic Values: '*'3667'

The attribute values "full3667", "noModification3667" and "noDerivatives3667" are similar to their counterparts above, except that they use text specified in RFC 3667 (February 2004).

A.2.2.4. Historic Values: '*'2026'

The attribute values "full2026" and "noDerivativeWorks2026" are similar to their counterparts above, except that they use text specified in RFC 2026 (October 1996).

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

Appendix B. Changes from RFC 2629 ('v1')

[[v1diff: See <http://greenbytes.de/tech/webdav/draft-reschke-xml2rfc-02.html#rfc.section.E> for a comparison of the schemata.]]

B.1. Removed Elements

The <appendix> element has been removed; to generate an appendix, place a <section> inside <back>.

B.2. Changed Defaults

Many attributes have lost their "default" value; this is to avoid having document semantics differ based on whether a DTD was specified and evaluated. Processors will handle absent values the way the default value was specified before.
B.3. Changed Elements

<artwork>: Has a set of new attributes: "name", "type", "src", "align", "alt", "width", and "height". (Section 2.5)

<author>: The <organization> element is now optional. The "role" attribute was added. (Section 2.6)

<country>: The requirement to use ISO 3166 codes was removed. (Section 2.11)

<date>: All attributes are now optional. (Section 2.13)

<figure>: Has a set of new attributes: "suppress-title", "src", "align", "alt", "width", and "height". (Section 2.17)

<iref>: Has a new "primary" attribute. (Section 2.20)

<list>: The "style" attribute isn’t restricted to a set of enumerated values anymore. The "hangIndent" and "counter" attributes have been added. (Section 2.22)

<rfc>: The "ipr" attribute has gained additional values. The attributes "consensus", "iprExtract", "submissionType", and "xml:lang" have been added. (Section 2.33)

<reference>: <annotation> allows adding prose to a reference. The "anchor" attribute has been made mandatory. (Section 2.30)

<references>: Can now appear multiple times, and carry a "title" attribute (so that normative and informative references can be split). (Section 2.31)

<section>: The new "toc" attribute controls whether it will appear in the Table Of Contents. <iref> can now appear as direct child element. (Section 2.34)

<t>: The "anchor" attribute can now be used as well, however there are restrictions on how they can be referred to. (Section 2.38)

B.4. New Elements

The following elements have been added: <annotation> (Section 2.3), <c> (Section 2.8), <cref> (Section 2.12), <format> (Section 2.18), <spanx> (Section 2.36), <texttable> (Section 2.39).

Appendix C. Relax NG Schema
namespace a = "http://relaxng.org/ns/compatibility/annotations/1.0"

rfc =
  element rfc {
    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 seriesNo { text }?,
    attribute ipr { "full2026"
      | "full13667"
      | "noDerivativeWorks2026"
      | "none"
      | "noModification3667"
      | "noDerivatives3667"
      | "full13978"
      | "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 }?,
    front, middle, back?
  }
front =
  element front {
    title, author+, date, area*, workgroup*, keyword*, abstract?,
    note*
  }
title =
  element title {
    attribute abbrev { text }?,
    "Reschke" Expires October 10, 2014 [Page 46]
text

author =
  element author {
    attribute initials { text }?,
    attribute surname { text }?,
    attribute fullname { text }?,
    attribute role { "editor" }?,
    organization?,
    address?
  }

organization =
  element organization {
    attribute abbrev { 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 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+ }
  note =
    element note {
      attribute title { text },
      t+
    }
  middle = element middle { section+ }
  section =

attribute anchor { xsd:ID }?,
attribute title { text },
    [ a:defaultValue = "default" ]
attribute toc { "include" | "exclude" | "default" }?,
(t | figure | texttable | iref)*,
section*
}

\text =
element t {
    attribute anchor { xsd:ID }?,
    attribute hangText { text }?,
    (text
        | \list
        | figure
        | xref
        | eref
        | iref
        | cref
        | spanx
        | vspace)*
}

\list =
element list {
    attribute style { text }?,
    attribute hangIndent { text }?,
    attribute counter { text }?,
    t+
}

xref =
element xref {
    attribute target { xsd:IDREF },
    [ a:defaultValue = "false" ] attribute pageno { "true" | "false"
},
    [ a:defaultValue = "default" ]
    attribute format { "counter" | "title" | "none" | "default" }?,
    text
}
eref =
element eref {
    attribute target { text },
    text
}
iref =
element iref {
    attribute item { text },
    [ a:defaultValue = "" ] attribute subitem { text }?,
    [ a:defaultValue = "false" ]
    attribute primary { "true" | "false" }?,

element cref {
    attribute anchor { xsd:ID }?,
    attribute source { text }?,
    text
}

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

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

element figure {
    attribute anchor { xsd:ID }?,
    [ 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 }?,
    iref*,
    preamble?,
    artwork,
    postamble?
}

element preamble { (text | xref | eref | iref | cref | spanx)* }

element artwork {
    [ a:defaultValue = "preserve" ]
    attribute xml:space { "default" | "preserve" }?,
    [ a:defaultValue = "" ] attribute name { text }?,
    [ a:defaultValue = "" ] attribute type { text }?,
    attribute src { text }?,
    [ a:defaultValue = "left" ]
    attribute align { "left" | "center" | "right" }?,

attribute value { text },
    empty
} format =
    element format {
        attribute target { text }?,
        attribute type { text },
        attribute octets { text }?,
        empty
    } annotation =
    element annotation { (text | xref | eref | iref | cref | spanx)* } start = rfc

(This schema was derived from version 1.3.6 of the xml2rfc DTD

Appendix D. Proposed Future Changes for ‘v3’

Discussion of "v3" changes takes place on the rfc-interest mailing
list (rfc-interest@rfc-editor.org), which has its home page at
<http://www.rfc-editor.org/mailman/listinfo/rfc-interest>. See also
a related Wiki page.

D.1. Contact Information

If contact information is changed to allow non-ASCII characters: add
a place for an ASCII fallback (probably just for the author names).

The content model for <postal> ought to be more strict to allow at
most one of <city>, <region>, <code>, and <country>.

It should be possible to have multiple <email> and <uri> elements
(see also <http://trac.tools.ietf.org/tools/xml2rfc/trac/ticket/36>).

<facsimile> looks outdated, while a container for IM (messaging) URIs
is missing. Maybe this area needs to be aligned with vCard.

Section 4.11 of [RFCSTYLE] hints at a "Contributors" Section that
could supply contact information similar to the one in the auto-
generated "Authors’ Address" Section. Consider how to capture
contributor contact information (probably not using <author> to avoid
confusion). Furthermore, consider ways to augment the contact
information section with prose.
D.2. Figures

Cleanup the set of overlapping attributes between <figure> and <artwork>.

For artwork that consists of a sequence of items (such as messages in a protocol example), it would be good if a <figure> element could contain multiple <artwork> elements (to assist code to find good places for page breaks).

Extend <figure> to support different types of artwork (such as by specifying certain type attribute values, see <http://greenbytes.de/tech/webdav/rfc2629xslt/rfc2629xslt.html#artwork.types>), and also avoid having to markup code (such as ABNF) as "artwork".

It would be good if "code components" could be marked as such.

Finally, even in preformatted text use of markup could be useful to support (a) references, or (b) highlighting the important bits (<http://greenbytes.de/tech/webdav/rfc2629xslt/rfc2629xslt.html#ext-rfc2629.artwork>).

D.3. Linking

Extend <xref> so that subsection/anchors can be specified (see <http://greenbytes.de/tech/webdav/rfc2629xslt/rfc2629xslt.html#ext-rfc2629.xref>).

Remove the "pageno" attribute which seems to be both undocumented and non-functional.

D.4. Lists

Allow multiple paragraphs in list items; eliminating the need to use <vspace> -- this could be achieved by adding a list item container element ("<lt>", see <http://greenbytes.de/tech/webdav/rfc2629xslt/rfc2629xslt.html#ext.element.lt> and <http://www.ietf.org/mail-archive/web/xml2rfc/current/msg02010.html>.

Add support for a "dictionary" style; eliminating the need to combine "hanging" with <vspace> to force new lines (see thread around <http://www.rfc-editor.org/pipermail/rfc-interest/2013-December/005876.html>).

D.5. References

Allow overriding the "anchor" attribute of an included <reference> element.

Add a way to add prose to a reference that avoids abuse of <seriesInfo>.

Allow <reference>s that identify a document set such as a BCP.

Deprecate or remove the <format> element; right now it’s not used for the generation of the plain text document anyway.

The "anchor" attribute is optional because it is not needed when using numeric references (symrefs processing instruction), and the reference actually is not in use. This is an edge case that doesn’t need special support in the vocabulary and thus should be removed.

D.6. Archival Considerations

When this vocabulary becomes the canonical RFC format, it will need to be able to capture all generated information, such as section/figure/table numbers, plus any auto-generated boilerplate (copyright statements etc.).

D.7. Document Metadata

Extend the concept of language tagging to at least examples and contact information to address potential japanese/chinese font confusion.

Provide a way to indicate the intended level on the standards track.

Include feedback information in a way so that generated documents can provide usable feedback links (see <http://greenbytes.de/tech/webdav/rfc2629xslt/rfc2629xslt.html#ext.element.feedback>).

D.8. Including Material

As discussed in Section 4, file inclusion currently uses out-of-the-box XML mechanisms or processor-specific directives.

We need to decide whether the vocabulary should have its own inclusion mechanism, or whether it would be better to use a generic solution such as [XINCLUDE] instead.

Make the <date> element optional; all of its content is optional already.

<spanx> has both a weird whitespace model ("preserve") and problematic styling. Consider to deprecate it in favor of elements such as <b>, <i>, and <tt>.

Indented paragraphs currently can be created by abusing the <list>. It would be good to have a special element for this purpose.

Provide a special element for inserting block quotes (<http://greenbytes.de/tech/webdav/rfc2629xslt/rfc2629xslt.html#ext.element.blockquote>).

The content model for <cref> should be extended to allow more flow elements, such as <xref> and <eref>.

Section titles should really be elements, not attributes (this would allow them to contain markup).

Text tables are currently very constrained. For instance, it would be good if alignment of headers and table cells could be de-coupled (<http://trac.tools.ietf.org/tools/xml2rfc/trac/ticket/69>).

Counters are currently restricted to lists, figures, and tables. Maybe there should be a generic mechanism that is not directly tied to other elements (<http://trac.tools.ietf.org/tools/xml2rfc/trac/ticket/68>).

Index

A
  abbrev attribute
    in organization element  20
    in title element  32
  abstract element  4
    inside front  16
  address element  4
    inside author  9
  align attribute
    in artwork element  7
    in figure element  13
    in texttable element  30
    in ttcol element  32
  alt attribute
    in artwork element  7
in figure element 14
anchor attribute
  in cref element 11
  in figure element 14
  in reference element 22
  in section element 27
  in t element 30
  in texttable element 31
annotation element 5
  inside reference 22
application/rfc+xml Media Type 37
area element 6
  inside front 15
artwork element 6
  align attribute 7
  alt attribute 7
  height attribute 7
  inside figure 13
  name attribute 7
  src attribute 7
  type attribute 7
  width attribute 8
xml:space attribute 8

Attributes
  abbrev 20, 32
  align 7, 13, 30, 32
  alt 7, 14
  anchor 11, 14, 22, 27, 30-31
  blankLines 33
  category 23
  consensus 24
  counter 17
  day 12
  docName 24
  format 34
  fullname 9
  hangIndent 17
  hangText 30
  height 7, 14
  initials 9
  ipr 24
  iprExtract 25
  item 16
  month 12
  name 7, 28
  number 25
  obsoletes 25
  octets 15
inside postal 21
cref element 11
   anchor attribute 11
   inside annotation 5
   inside c 10
   inside postamble 21
   inside preamble 21
   inside t 29
   source attribute 11

date element 11
day attribute 12
   inside front 15
   month attribute 12
   year attribute 12
day attribute
   in date element 12
docName attribute
   in rfc element 24

Elements
abstract 4, 16
address 4, 9
annotation 5, 22
area 6, 15
artwork 6, 13
author 8, 15
back 9, 23
c 10, 30
city 10, 20
code 10, 20
country 10, 21
cref 5, 10-11, 21, 29
date 11, 15
e-mail 5, 12
eref 5, 10, 12, 21, 29
facsimile 5, 13
figure 13, 27, 29
format 15, 22
front 15, 22-23
iref 5, 10, 13, 16, 21, 27, 29
keyword 16-17
list 17, 29
middle 19, 23
note 16, 19
organization 9, 20
octets attribute 15
target attribute 15
type attribute 15
front element 15
  inside reference 22
  inside rfc 23
fullname attribute
  in author element 9

H
  hangIndent attribute
    in list element 17
  hangText attribute
    in t element 30
  height attribute
    in artwork element 7
    in figure element 14

I
  initials attribute
    in author element 9
  ipr attribute
    '*2026' 44
    '*3667' 44
    '*3978' 44
    '*trust200811' 44
    '*trust200902' 42
    'noDerivativesTrust200902' 43
    'noModificationTrust200902' 43
    'pre5378Trust200902' 43
    'trust200902' 42
    in rfc element 24
  iprExtract attribute
    in rfc element 25
  iref element 16
    inside annotation 5
    inside c 10
    inside figure 13
    inside postamble 21
    inside preamble 21
    inside section 27
    inside t 29
    item attribute 16
    primary attribute 16
    subitem attribute 16
    item attribute
      in iref element 16
K
  keyword element 17
  inside front 16

L
  list element 17
  counter attribute 17
  hangIndent attribute 17
  inside t 29
  style attribute 18
  list styles
  empty 18
  format ... 18
  hanging 18
  letters 18
  numbers 18
  symbols 18

M
  Media Type
    application/rfc+xml 37
  middle element 19
    inside rfc 23
  month attribute
    in date element 12

N
  name attribute
    in artwork element 7
    in seriesInfo element 28
  note element 19
    inside front 16
    title attribute 19
  number attribute
    in rfc element 25

O
  obsoletes attribute
    in rfc element 25
  octets attribute
    in format element 15
  organization element 20
    abbrev attribute 20
    inside author 9

P
  pageno attribute
    in xref element 35
name attribute 28
value attribute 28
seriesNo attribute
in rfc element 25
source attribute
in cref element 11
spanx element 28
inside annotation 5
inside c 10
inside postamble 21
inside preamble 21
inside t 29
style attribute 28
xml:space attribute 28
src attribute
in artwork element 7
in figure element 14
street element 29
inside postal 20
style attribute
in list element 18
in spanx element 28
in texttable element 31
subitem attribute
in iref element 16
submissionType attribute
in rfc element 26
suppress-title attribute
in figure element 14
in texttable element 31
surname attribute
in author element 9

T
t element 29
anchor attribute 30
hangText attribute 30
inside abstract 4
inside list 17
inside note 19
inside section 27
target attribute
in eref element 13
in format element 15
in reference element 22
in xref element 35
texttable element 30
align attribute 30
anchor attribute 31
inside section 27
style attribute 31
suppress-title attribute 31
title attribute 32
title attribute
in figure element 14
in note element 19
in references element 23
in section element 27
in texttable element 32
title element 32
abbrev attribute 32
inside front 15
toc attribute
in section element 27
ttcoll element 32
align attribute 32
inside texttable 30
width attribute 32
type attribute
in artwork element 7
in format element 15

U
updates attribute
in rfc element 26
uri element 33
inside address 5

V
value attribute
in seriesInfo element 28
vspace element 33
blankLines attribute 33
inside t 29

W
width attribute
in artwork element 8
in figure element 14
in ttcol element 32
workgroup element 33
inside front 15

X
xml:lang attribute
in rfc element 26
xml:space attribute
  in artwork element  8
  in spanx element  28
xref element  34
  format attribute  34
  inside annotation  5
  inside c  10
  inside postamble  21
  inside preamble  21
  inside t  29
  pageno attribute  35
  target attribute  35

xref formats
  counter  34
  default  34
  none  34
  title  34

Y
  year attribute
  in date element  12

Author’s Address

  Julian F. Reschke
  greenbytes GmbH
  Hafenweg 16
  Muenster, NW  48155
  Germany

EMail: julian.reschke@greenbytes.de
URI:    http://greenbytes.de/tech/webdav/