The 'XML2RFC' version 2 Vocabulary
draft-reschke-xml2rfc-09

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>.

to this document.

Table of Contents

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", [V1rev]). Furthermore, it discusses potential extensions in a
future revision ("v3").

Processing Instructions (Section 2.6 of [XML]) generally are specific
to a given processor, and thus are not considered to be part of the
vocabulary. See Section 4.1 of [TCLReadme] for a list of the
processing instructions supported by the first implementation of an
xml2rfc processor.

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.

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:

```
<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:

o   Text

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](#))

[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 fullname or the abbreviation of one
   of the IETF areas as listed on <http://www.ietf.org/iesg/area.html>.
   The list at the time that this document is being published is:
   "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:

   o diagrams ("line art"),

   o source code,

   o formal languages (such as ABNF or the RNC notation used in this
      document),

   o complex tables, or

   o protocol unit diagrams.

   Alternatively, the "src" attribute allows referencing an external
   graphics file, such as a bitmap or a vector drawing, using a URI. 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:

o "left" (default)

o "center"

o "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 is wholly part of the XML file.

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:

o "default"

o "preserve" (default)

2.6. <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 (inside of <reference>).

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:

o "editor"

2.6.4. 'surname' attribute

The author's surname.

2.7. <back>

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

o Text

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)

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](#)).

Content model: only text content.

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](#)]),
additionally constrained to US-ASCII characters ([[USASCII](#)]).

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 boilerplate 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 one or more of "year",
"month", or "day" are left out, the processor will attempt to use the
current system date if the attributes that are present are consistent
with that date.

Note that in this case, 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 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 child element of: <front> (Section 2.19).

Content model: this element does not have any contents.

2.13.1. 'day' attribute

Day of publication; this is a number.

2.13.2. 'month' attribute

Month of publication; this is the English name of the month.

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:

o "left" (default)

o "center"

o "right"

[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:

o "true"

o "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 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:

<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:

   o "true"

   o "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 documents.

   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"

   For unlabeled list items; it can also be used for indentation
   purposes (this is the default value when there is an enclosing
   list where the style is specified).

"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 (see [RFCSTYLE], Section 4.1.1 for more
information). 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:

o Text

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)

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:

o Text

   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](#))

[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](#)]),
   additionally constrained to US-ASCII characters ([[USASCII](#)]).

[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" as described
in 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)

2.31.1. '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:

o "std"

o "bcp"

o "info"

o "exp"

o "historic"

2.33.2. 'consensus' attribute

Affects the generated boilerplate.

See [RFC5741] for more information.

Allowed values:

o "no"

o "yes"

2.33.3. 'docName' attribute

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

A processor should give an error if both the "docName" and "number"
attributes are given in the <rfc> element.

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:

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"

2.33.5. '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.

A processor should give an error if both the "docName" and "number" attributes are given in the <rfc> element.

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:

o "IETF" (default)

o "IAB"

o "IRTF"

o "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:

o "include"

o "exclude"

o "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: ([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:

o "left"

o "center" (default)

o "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

o "all" means borders around all table cells,

o "full" is like "all" except no horizontal lines between table rows
   (except below the column titles),

o "headers" adds just a separator between column titles and rows,
   and

o "none" means no borders at all.

Allowed values:

o "all"

o "none"

o "headers"

o "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:

o "true"

o "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 specify
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:

o "left" (default)

o "center"

o "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 (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 2.33.9).

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:

o "counter"

o "title"

o "none"

o "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:

o "true"

o "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. Escaping for Use in XML

Text in XML cannot use the literal characters "<" and "&", as they
have special meaning to the XML processor (starting entities,
elements, etc.). Usually, these characters will need to be
substituted by "<" and "&" (see Section 4.6 of [XML]).

">" does not require escaping, unless it appears in the sequence
"]]>" (which indicates the end of a CDATA section, see below).

Escaping the indivual characters can be a lot of work (when done
manually), and also messes up alignment in artwork. Another approach
to escaping is to use CDATA sections ([XML], Section 2.7). Within
these, no further escaping is needed, except when the "end-of-CDATA"
marker needs to be used (in that case, the CDATA section needs to be
closed, and a new one needs to be started).

4. 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 frequenly used in titles (by excluding certain space
   characters from the line breaking algorithm, the processor will
   use the remaining whitespace ocurrences 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 this:

```
<?xml version="1.0"?>

<!DOCTYPE rfc [

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

5. 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] and Section 4.1.2 of
[TCLReadme]).

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.

6. 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
characters ([USASCII] and Section 3 of [RFC2223]). Future versions
are likely to relax this rule, and it is expected that the vocabulary
will be extended so that US-ASCII alternatives can be provided when
that makes sense (for instance, in contact information).

7. 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, the nature of XML, plus vocabulary features such as
typed artwork, make it attractive to extract content from documents
for further processing, such for the purpose of checking syntax, or
computing/verifying examples. In the latter case, care needs to be
taken that only trusted content is processed.

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

8. IANA Considerations

8.1. 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 7. 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 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 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: N/A

   Author: See Authors Section.

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

9. 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.

10. References

10.1. 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,
               <http://www.w3.org/TR/2008/REC-xml-20081126/>.

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

10.2. Informative References

[BCP13]   Freed, N., Klensin, J., and T. Hansen, "Media Type
               Specifications and Registration Procedures", BCP 13,
               RFC 6838, January 2013.

[IDGUIDE]   Housley, R., "Guidelines to Authors of Internet-Drafts",
               December 2010,
               <http://www.ietf.org/id-info/guidelines.html>.

[JING]   Thai Open Source Software Center Ltd, "Jing - A RELAX NG
               validator in Java", 2008,
               <http://www.thaiopensource.com/relaxng/jing.html>.

               Downloads:
               <https://code.google.com/p/jing-trang/downloads/list>.

[RFC2026]   Bradner, S., "The Internet Standards Process -- Revision
               3", BCP 9, RFC 2026, October 1996.

[RFC2046]   Freed, N. and N. Borenstein, "Multipurpose Internet Mail
               Extensions (MIME) Part Two: Media Types", RFC 2046,
               November 1996.

[RFC2223]   Postel, J. and J. Reynolds, "Instructions to RFC
               Authors", RFC 2223, October 1997.

[RFC2397]   Masinter, L., "The "data" URL scheme", RFC 2397,
               August 1998.

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

   [RFC3023]   Murata, M., St. Laurent, S., and D. Kohn, "XML Media
                Types", RFC 3023, January 2001.

   [RFC3470]   Hollenbeck, S., Rose, M., and L. Masinter, "Guidelines
                for the Use of Extensible Markup Language (XML) within
                IETF Protocols", BCP 70, RFC 3470, January 2003.

   [RFC3966]   Schulzrinne, H., "The tel URI for Telephone Numbers",
                RFC 3966, December 2004.

   [RFC3986]   Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
                Resource Identifier (URI): Generic Syntax", STD 66,
                RFC 3986, January 2005.

   [RFC5598]   Crocker, D., "Internet Mail Architecture", RFC 5598,
                July 2009.

                PDF version: <http://www.rfc-editor.org/rfc/rfc5598.pdf>

   [RFC5741]   Daigle, L. and O. Kolkman, "RFC Streams, Headers, and
                Boilerplates", RFC 5741, December 2009.

   [RFC6068]   Duerst, M., Masinter, L., and J. Zawinski, "The 'mailto'
                URI Scheme", RFC 6068, October 2010.

   [RFC6266]   Reschke, J., "Use of the Content-Disposition Header
                Field in the Hypertext Transfer Protocol (HTTP)",
                RFC 6266, June 2011.

   [RFCPOLICY] RFC Editor, "RFC Editorial Guidelines and Procedures",
                February 2012, <http://www.rfc-editor.org/policy.html>.

   [RFCSTYLE]   Heather, H. and S. Ginoza, "RFC Style Guide",
                draft-iab-styleguide-02 (work in progress), April 2014.

   [RNC]   Clark, J., "RELAX NG Compact Syntax", OASIS ,
                November 2002, <http://www.oasis-open.org/committees/
                relax-ng/compact-20021121.html>.

   [TCLReadme] Rose, M., Fenner, B., and C. Levert, "xml2rfc
                v1.35pre1", October 2009, <http://svn.tools.ietf.org/
                svn/tools/xml2rfc/archive/README.html>.

   [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>.

   [UNICODE]   The Unicode Consortium, "The Unicode Standard, Version
                    6.3.0", September 2013,
                    <http://www.unicode.org/versions/Unicode6.3.0/>.

   [USASCII]   American National Standards Institute, "Coded Character
                    Set -- 7-bit American Standard Code for Information
                    Interchange", ANSI X3.4, 1986.

   [V1rev]   Rose, M., "Writing I-Ds and RFCs using XML (revised)",
                    February 2008, <http://svn.tools.ietf.org/svn/tools/
                    xml2rfc/archive/draft-mrose-writing-rfcs.html>.

   [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/2006/REC-xinclude-20061115/>.

                    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 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 "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"
      | "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 }?,
    front,
    middle,
    back?
  }
```

```
front =
  element front {
    title, author+, date, area*, workgroup*, keyword*, abstract?,
note*
  }
title =
  element title {
    attribute abbrev { text }?,
    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 =
  element section {
    attribute anchor { xsd:ID }?,
    attribute title { text },
    [ a:defaultValue = "default" ]
    attribute toc { "include" | "exclude" | "default" }?,
    (t | figure | texttable | iref)*,
    section*
  }
t =
  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" }?,
        empty
      }
    cref =
      element cref {
        attribute anchor { xsd:ID }?,
        attribute source { text }?,
        text
      }
    spanx =
      element spanx {
        [ a:defaultValue = "preserve" ]
        attribute xml:space { "default" | "preserve" }?,
        [ a:defaultValue = "emph" ] attribute style { text }?,
        text
      }
    vspace =
      element vspace {
        [ a:defaultValue = "0" ] attribute blankLines { text }?,
        empty
      }
    figure =
      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?
      }
    preamble =
      element preamble { (text | xref | eref | iref | cref | spanx)* }
    artwork =
```

```
    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" }?,
      [ a:defaultValue = "" ] attribute alt { text }?,
      [ a:defaultValue = "" ] attribute width { text }?,
      [ a:defaultValue = "" ] attribute height { text }?,
      text*
    }
  postamble =
    element postamble { (text | xref | eref | iref | cref | spanx)* }
  texttable =
    element texttable {
      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" }?,
      preamble?,
      ttcol+,
      c*,
      postamble?
    }
  ttcol =
    element ttcol {
      attribute width { text }?,
      [ a:defaultValue = "left" ]
      attribute align { "left" | "center" | "right" }?,
      text
    }
  c = element c { (text | xref | eref | iref | cref | spanx)* }
  back = element back { references*, section* }
  references =
    element references {
      [ a:defaultValue = "References" ] attribute title { text }?,
      reference+
    }
  reference =
    element reference {
      attribute anchor { xsd:ID },
      attribute target { text }?,
```

```
      front,
      seriesInfo*,
      format*,
      annotation*
    }
  seriesInfo =
    element seriesInfo {
      attribute name { text },
      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
('Document Type Definition', [XML], Section 2.8), available from <htt
p://svn.tools.ietf.org/svn/tools/xml2rfc/vocabulary/v2/03/
xml2rfcv2.dtd>).

C.1. Checking Validity

The validity of XML files can be checked with any tool that supports
Relax NG ([RNC]). The reference implementation is the Java-based,
open sourced "JING" ([JING]).

To use JING, download the latest ZIP file from the downloads page
(currently <https://code.google.com/p/jing-trang/downloads/
detail?name=jing-20091111.zip>), extract the archive, copy "jing.jar"
from the "bin" folder, and make sure Java is installed).

To check a file "test.xml" using the RNC file "schema.rnc", run (from
a command line prompt):

java -jar jing.jar -c schema.rnc test.xml

In good Unix tradition, no output means the file is valid.

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
   <https://www.rfc-editor.org/rse/wiki/doku.php?id=design:xml-tags> for
   a related Wiki page.

D.1. Contact Information

   If contact information is changed to allow non-ASCII characters: add
   a place for a 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 5, 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.

D.9. Misc.

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

      N
         name attribute
            in artwork element 7
            in seriesInfo element 28
         note element 19
            inside front 16
            title attribute 20
         number attribute
            in rfc element 26

      O
         obsoletes attribute
            in rfc element 26
         octets attribute
            in format element 15
         organization element 20
            abbrev attribute 20
            inside author 9

      P
         pageno attribute
            in xref element 35
         phone element 20
            inside address 5
         postal element 20
            inside address 5
         postamble element 21
            inside figure 14
            inside texttable 31
         preamble element 21
            inside figure 13
            inside texttable 31
         primary attribute
            in iref element 16

      R
         reference element 22
            anchor attribute 22
            inside references 23
            target attribute 22
         references element 23
            inside back 10
            title attribute 23
         region element 23
            inside postal 21
         rfc element 23

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/