

Transforming RFC7749-formatted XML through XSLT

Table of Contents

1	Introduction.....	7
2	Supported RFC7749 elements.....	8
2.1	Extension elements.....	8
3	Processing Instructions.....	9
3.1	Supported xml2rfc-compatible PIs.....	9
3.2	Unsupported xml2rfc-compatible PIs.....	10
3.3	Extension PIs.....	10
3.3.1	abort-on - Log Level.....	10
3.3.2	allow-markup-in-artwork - Allow certain XML elements inside <artwork> and <sourcecode>.....	11
3.3.3	authors-section - Placement of 'Authors' Section.....	11
3.3.4	css-contents - CSS Contents.....	11
3.3.5	css-resource - Custom CSS Resource.....	11
3.3.6	dark-mode - Switch for Disabling 'Dark Mode' Support.....	11
3.3.7	diff-uri - URI Template for Internet Draft Diff Links.....	11
3.3.8	doi-uri - URI Template for DOI Links.....	12
3.3.9	duplex - Support Duplex Printing.....	12
3.3.10	errata - Embed Errata Information.....	12
3.3.11	html-pretty-print - Switch for Enabling Pretty Printing of Code.....	12
3.3.12	include-generator - Switch for Disabling Generator Information in Output.....	12
3.3.13	include-index - Switch for Disabling Index Generation.....	13
3.3.14	include-references-in-index - Generate Index Entries for References.....	13
3.3.15	insert-metadata - Dynamic Metadata Insertion in HTML.....	13
3.3.16	internet-draft-base-uri - base URI for RFC reference files.....	13
3.3.17	internet-draft-uri - URI Template for Interned Drafts.....	13
3.3.18	justification - Text Justification.....	13
3.3.19	isbn-uri - URI Template for ISBNs.....	14
3.3.20	log-level - Log Level.....	14
3.3.21	maxwidth - Maximal Text Width in HTML.....	14
3.3.22	parse-xml-in-artwork - Parse and Check XML in artwork.....	14
3.3.23	rfc-errata-uri - URI Template for RFC Errata.....	14
3.3.24	rfc-erratum-uri - URI Template for a specific RFC Erratum.....	15
3.3.25	rfc-reference--base-uri - base URI for RFC reference files.....	15
3.3.26	rfc-uri - URI Template for HTML Version of RFCs.....	15
3.3.27	sec-no-trailing-dots - Disable Trailing Dots in Section Numbers.....	15
3.3.28	std-uri - URI Template for IETF Standards.....	15
3.3.29	support-highwire-press-tags - Support 'Highwire Press' Metadata.....	15
3.3.30	support-open-graph-tags - Support 'Highwire Press' Metadata.....	16
3.3.31	support-rfc2731 - Support RFC 2731 Metadata.....	16
3.3.32	ucd-file - Filename of Unicode Database.....	16
3.3.33	xml2rfc-backend - Select xml2rfc Target Version.....	16
3.3.34	xref-with-text-generate - Default Handling of <xref> with Text Content.....	16
4	Anchors.....	18
5	Supported XSLT engines.....	19
5.1	Standalone Engines.....	19
5.2	In-Browser Engines.....	19

6 Transforming to HTML.....	21
6.1 HTML compliance.....	21
6.2 Standard HTML LINK elements.....	22
6.3 Standard HTML metadata.....	23
6.4 Dublin Core (RFC2731) metadata.....	23
6.5 Insertion of Live RFC Meta Data.....	23
6.6 Insertion of RFC Errata Links.....	24
7 Transforming to XHTML.....	25
8 Transforming to CHM (Microsoft Compiled Help).....	26
9 Transforming to PDF.....	27
9.1 Via XSL-FO.....	27
9.1.1 Example: producing output for Apache FOP.....	27
9.2 Via HTML.....	27
10 Transforming to ePub.....	28
11 Generic Extensions.....	29
11.1 <abnf-char-sequence> element.....	29
11.2 <anchor-alias> element.....	29
11.3 <bb> element.....	29
11.4 <bc> element.....	29
11.5 <bcp14> element.....	29
11.6 <blockquote> element.....	30
11.7 <bt> element.....	30
11.8 <c> element.....	30
11.9 <contributor> element.....	30
11.10 <dfn> element.....	31
11.11 <feedback> element.....	31
11.12 <h> element.....	31
11.13 <highlight> element.....	31
11.14 <length-of> element.....	31
11.15 <link> element.....	31
11.16 <lt> element.....	32
11.17 <note> element.....	32
11.18 <parse-xml> element.....	32
11.19 <prose> element.....	32
11.20 <q> element.....	32
11.21 <ref> element.....	32
11.22 <source> element.....	33

11.23	<sup> element.....	33
11.24	<u-map> element.....	33
11.25	Extensions to <artwork> element.....	33
11.26	Extensions to <date> element.....	34
11.27	Extensions to <eref> element.....	34
11.28	Extensions to <iref> element.....	34
11.29	Extensions to <list> element.....	34
11.30	Extensions to <rfc> element.....	34
11.31	Extensions to <section> element.....	34
11.32	Extensions to <xref> element.....	35
12	Experimental Support for RFC7991bis Vocabulary.....	36
12.1	artset Element.....	36
12.2	aside Element.....	36
12.3	bcp14 Element.....	36
12.4	blockquote Element.....	36
12.5	boilerplate Element.....	36
12.6	br Element.....	36
12.7	cityarea Element.....	36
12.8	contact Element.....	36
12.9	displayreference Element.....	36
12.10	dd Element.....	36
12.11	dl Element.....	36
12.12	dt Element.....	36
12.13	em Element.....	37
12.14	extaddr Element.....	37
12.15	li Element.....	37
12.16	link Element.....	37
12.17	name Element.....	37
12.18	ol Element.....	37
12.19	pobox Element.....	37
12.20	postalLine Element.....	37
12.21	refcontent Element.....	37
12.22	Extensions to reference Element.....	37
12.22.1	quoteTitle attribute.....	37
12.23	referencegroup Element.....	37
12.24	relref Element.....	37
12.25	Extensions to section Element.....	38
12.25.1	numbered attribute.....	38
12.25.2	removeInRFC attribute.....	38

12.26	sortingcode Element.....	38
12.27	sourcecode Element.....	38
12.28	strong Element.....	38
12.29	sub Element.....	38
12.30	sup Element.....	38
12.31	svg Element.....	38
12.32	table Element.....	38
12.33	tbody Element.....	38
12.34	td Element.....	38
12.35	tfoot Element.....	38
12.36	th Element.....	39
12.37	thead Element.....	39
12.38	tr Element.....	39
12.39	tt Element.....	39
12.40	u Element.....	39
12.41	ul Element.....	39
12.42	Extensions to xref Element.....	39
12.42.1	relative attribute.....	39
12.42.2	section attribute.....	39
12.42.3	sectionFormat attribute.....	39
12.43	xi:include Element.....	39
13	Utilities.....	40
13.1	Checking References.....	40
13.2	Generating Graphs from References.....	41
13.3	Producing reference entries for books.....	42
13.4	Down-converting to RFC 7749 Grammar.....	43
13.5	Extracting artwork.....	44
13.6	GRRDL.....	44
13.7	HTML Live Refresh.....	44
13.8	Refreshing included material in the XML source.....	45
14	Informative References.....	47
Appendix A	RELAX NG Compact Schema.....	49
Appendix B	Implementation Notes.....	51
B.1	Recognized type attributes for <artwork> element.....	51
Appendix C	Examples.....	52
C.1	Using the 'Internal Subset'.....	52
C.2	Customization.....	52

Appendix D Producing the IETF 'Boilerplate'	54
Appendix E License	55
Index	56
Author's Address	62

1. Introduction

This document describes a set of XSLT transformations that can be used to transform "XML2RFC" XML ([RFC7749], updating [RFC2629]) to various output formats, such as HTML and PDF. The main topics are

- compliance to the xml2rfc XML element set (Section 2),
- support for xml2rfc processing instructions (Section 3),
- the names of anchor elements generated in HTML and PDF output (Section 4),
- various XSLT engines that can be used (Section 5),
- outputting HTML (Section 6) and XHTML (Section 7),
- outputting CHM (Compiled Microsoft Help, Section 8),
- outputting PDF (Section 9),
- outputting ePub (Section 10),
- extensions to the xml2rfc vocabulary (Section 11),
- selected extensions from the proposed xml2rfc V3 vocabulary (Section 12),
- various utilities (Section 13).

The full distribution is available at <<https://greenbytes.de/tech/webdav/rfc2629xslt.zip>>. A mirror of the non-public source repository can be found at <<https://github.com/reschke/xml2rfc>>; this is also a good place for reporting issues.

Note: RFC 2629 was the initial specification of the XML vocabulary; that's why the "2629" lives on in various parts, such as filenames.

2. Supported RFC7749 elements

`rfc2629.xslt` supports both all grammar elements defined in [\[RFC7749\]](#), plus a subset of the new elements defined in [\[RFC7991bis\]](#).

2.1. Extension elements

`rfc2629.xslt` supports two kind of extension elements, using different XML namespaces.

The first set contains (hopefully) generally useful extensions, see [Section 11](#).

The second set is used for change and issue tracking and currently is not documented here. Please email the author in case you're interested in using these extensions.

3. Processing Instructions

All PIs can be set as XSLT parameter as well, overriding any value that is found in the source file to be transformed.

Using processing instructions:

```
<?rfc toc="yes"?>
<?rfc-ext support-rfc2731="no"?>
```

Using XSLT parameters (Saxon):

```
java -cp saxon.jar com.icl.saxon.StyleSheet source.xml rfc2629.xslt \
  xml2rfc-toc=yes xml2rfc-ext-support-rfc2731=no > result.html
```

Using XSLT parameters (xsltproc):

```
xsltproc --param xml2rfc-toc '"yes"' \
  --param xml2rfc-ext-support-rfc2731 '"no"' \
  rfc2629.xslt source.xml > result.html
```

(note the required quoting of string parameters)

3.1. Supported xml2rfc-compatible PIs

PI target	PI pseudo-attribute	XSLT parameter name	default	comment
rfc	authorship	xml2rfc-authorship	"yes"	when set to "no", the "Authors" section is suppressed
rfc	background	xml2rfc-background	(not set)	
rfc	compact	xml2rfc-compact	"no"	only applies to HTML output method when printing
rfc	comments	xml2rfc-comments	"no" ("yes" for v3 documents)	
rfc	docmapping		"yes"	This is the default for rfc2629.xslt anyway, and it can not be changed
rfc	editing	xml2rfc-editing	"no"	
rfc	footer	xml2rfc-footer	(not set)	
rfc	header	xml2rfc-header	(not set)	
rfc	include			only partly supported, use external entities instead (see Appendix C.1) or other tools (Section 13.8) instead
rfc	inline	xml2rfc-inline	"no" ("yes" for v3 documents)	

PI target	PI pseudo-attribute	XSLT parameter name	default	comment
rfc rfc rfc	iprnotified linkmailto multiple-initials	xml2rfc-iprnotified xml2rfc-linkmailto xml2rfc-multiple-initials	"no" "yes" "no"	determines whether the processor will attempt to truncate multiple initials to a single one; can be set globally (affecting the front page) but also as child element of <reference>
rfc rfc	private refparent	xml2rfc-private xml2rfc-private	(not set) "References"	Title for References sections when automatically inserted (limited support)
rfc rfc rfc	rfcedstyle sortrefs symrefs	xml2rfc-rfcedstyle xml2rfc-sortrefs xml2rfc-symrefs	(not set) "no" "yes"	The default has changed from "no" to "yes" as of June 6, 2007 and xml2rfc 1.33pre4.
rfc	toc	xml2rfc-toc	"no" ("yes" for documents specifying "3" as vocabulary version)	
rfc rfc	tocdepth topblock	xml2rfc-tocdepth xml2rfc-topblock	99 "yes"	

3.2. Unsupported xml2rfc-compatible PIs

PI target	PI pseudo-attribute	comment
rfc rfc rfc rfc rfc rfc	needLines slides strict subcompact tocindent toccompact	(defaults to "yes")

3.3. Extension PIs

3.3.1. abort-on - Log Level

PI target	rfc-ext
PI pseudo-attribute	abort-on
XSLT parameter name	xml2rfc-ext-abort-on
default	"OFF"

Controls at which log level a message causes the XSLT to be aborted (one of "OFF", "FATAL", "ERROR", "WARNING", "INFO", "DEBUG", "TRACE").

3.3.2. allow-markup-in-artwork - Allow certain XML elements inside <artwork> and <sourcecode>

PI target	rfc-ext
PI pseudo-attribute	allow-markup-in-artwork
XSLT parameter name	xml2rfc-ext-allow-markup-in-artwork
default	"no"

Enables support for specific elements inside <artwork> and <sourcecode> elements (using this extension makes the document incompatible to the RFC7749 grammar; see description of conversion XSLT in [Section 13.4](#)).

3.3.3. authors-section - Placement of 'Authors' Section

PI target	rfc-ext
PI pseudo-attribute	authors-section
XSLT parameter name	xml2rfc-ext-authors-section
default	"end"

When "before-appendices", place the authors section between references and appendices (this ordering was used a long time ago).

3.3.4. css-contents - CSS Contents

PI target	rfc-ext
PI pseudo-attribute	css-contents
XSLT parameter name	xml2rfc-ext-css-contents
default	none

CSS content to use instead of the built-in (experimental).

3.3.5. css-resource - Custom CSS Resource

PI target	rfc-ext
PI pseudo-attribute	css-resource
XSLT parameter name	xml2rfc-ext-css-resource
default	none

Name of CSS resource (URI or relate reference) to use instead of the built-in (experimental).

3.3.6. dark-mode - Switch for Disabling 'Dark Mode' Support

PI target	rfc-ext
PI pseudo-attribute	dark-mode
XSLT parameter name	xml2rfc-ext-dark-mode
default	"auto"

Set to 'no' to disable "dark mode" CSS support.

3.3.7. diff-uri - URI Template for Internet Draft Diff Links

PI target	rfc-ext
PI pseudo-attribute	diff-uri
XSLT parameter name	xml2rfc-ext-diff-uri
default	"https://www.ietf.org/rfcdiff?url2={ internet-draft }"
URI template for Internet Draft Diff links.	

3.3.8. doi-uri - URI Template for DOI Links

PI target	rfc-ext
PI pseudo-attribute	doi-uri
XSLT parameter name	xml2rfc-ext-doi-uri
default	"https://dx.doi.org/{ doi }"
URI template for DOIs links.	

3.3.9. duplex - Support Duplex Printing

PI target	rfc-ext
PI pseudo-attribute	duplex
XSLT parameter name	xml2rfc-ext-duplex
default	"no"
When set to "yes", format printed output for doublesided printing.	

3.3.10. errata - Embed Errata Information

PI target	rfc-ext
PI pseudo-attribute	errata
XSLT parameter name	xml2rfc-ext-errata
default	none

Can be used to specify an errata file; output will link to individual errata when possible. See [Section 6.6](#).

3.3.11. html-pretty-print - Switch for Enabling Pretty Printing of Code

PI target	rfc-ext
PI pseudo-attribute	html-pretty-print
XSLT parameter name	xml2rfc-ext-html-pretty-print
default	none

Used to specify a JS-based code pretty-printer; the value is the CSS class name to insert, followed by a blank space, followed by the URI of the JS library. For instance: "prettyprint https://cdn.rawgit.com/google/code-prettify/master/loader/run_prettify.js"

3.3.12. include-generator - Switch for Disabling Generator Information in Output

PI target	rfc-ext
PI pseudo-attribute	include-generator

XSLT parameter name	xml2rfc-ext-include-generator
default	"yes"

Set to 'no' in order to disable inclusion of generator version information.

3.3.13. include-index - Switch for Disabling Index Generation

PI target	rfc-ext
PI pseudo-attribute	include-index
XSLT parameter name	xml2rfc-ext-include-index
default	"yes"

When set to "no", no index will be generated.

3.3.14. include-references-in-index - Generate Index Entries for References

PI target	rfc-ext
PI pseudo-attribute	include-references-in-index
XSLT parameter name	xml2rfc-ext-include-references-in-index
default	"no"

When set to "yes", index entries are generated for all references.

3.3.15. insert-metadata - Dynamic Metadata Insertion in HTML

PI target	rfc-ext
PI pseudo-attribute	insert-metadata
XSLT parameter name	xml2rfc-ext-insert-metadata
default	"yes"

When set to "yes", include JS code that fetches current RFC/Internet-Draft metadata and inserts it into the front page (standards track, obsolescence, updates, errata, freshness of draft...).

3.3.16. internet-draft-base-uri - base URI for RFC reference files

PI target	rfc-ext
PI pseudo-attribute	internet-draft-reference-base-uri
XSLT parameter name	xml2rfc-ext-internet-draft-reference-base-uri
default	"https://bib.ietf.org/public/rfc/bibxml-ids/"

Base URI for Internet-Draft references included using the "include" processing instruction.

3.3.17. internet-draft-uri - URI Template for Interned Drafts

PI target	rfc-ext
PI pseudo-attribute	internet-draft-uri
XSLT parameter name	xml2rfc-ext-internet-draft-uri
default	"https://datatracker.ietf.org/doc/html/{internet-draft}"

URI template for Internet-Draft links (can be set locally inside the <reference> element as well to override the default).

3.3.18. justification - Text Justification

PI target	rfc-ext
PI pseudo-attribute	justification
XSLT parameter name	xml2rfc-ext-justification
default	"never"

"never": never emit justified text, "always": always emit justified text, "print": only emit justified text for print media.

3.3.19. isbn-uri - URI Template for ISBNs

PI target	rfc-ext
PI pseudo-attribute	isbn-uri
XSLT parameter name	xml2rfc-ext-isbn-uri
default	"https://www.worldcat.org/search?q=isbn:{ isbn }"

URI template for ISBN lookup.

3.3.20. log-level - Log Level

PI target	rfc-ext
PI pseudo-attribute	log-level
XSLT parameter name	xml2rfc-ext-log-level
default	"WARNING"

Logging level, one of "OFF", "FATAL", "ERROR", "WARNING", "INFO", "DEBUG", "TRACE".

3.3.21. maxwidth - Maximal Text Width in HTML

PI target	rfc-ext
PI pseudo-attribute	maxwidth
XSLT parameter name	xml2rfc-ext-maxwidth
default	1000

For HTML output: maximal text width in CSS pixels.

3.3.22. parse-xml-in-artwork - Parse and Check XML in artwork

PI target	rfc-ext
PI pseudo-attribute	parse-xml-in-artwork
XSLT parameter name	xml2rfc-ext-parse-xml-in-artwork
default	"no"

May be used to enable parsing of XML content in figures (MSXML only).

3.3.23. rfc-errata-uri - URI Template for RFC Errata

PI target	rfc-ext
PI pseudo-attribute	rfc-errata-uri
XSLT parameter name	xml2rfc-ext-rfc-errata-uri

default "https://www.rfc-editor.org/errata/rfc{rfc}"
 URI template for all RFC Errata for a given RFC ("rfc" is the RFC number).

3.3.24. rfc-erratum-uri - URI Template for a specific RFC Erratum

PI target rfc-ext
 PI pseudo-attribute rfc-erratum-uri
 XSLT parameter name xml2rfc-ext-rfc-erratum-uri
 default "https://www.rfc-editor.org/errata/eid{eid}"
 URI template for a specific RFC erratum ("eid" is the "errata id").

3.3.25. rfc-reference--base-uri - base URI for RFC reference files

PI target rfc-ext
 PI pseudo-attribute rfc-reference-base-uri
 XSLT parameter name xml2rfc-ext-rfc-reference-base-uri
 default "https://bib.ietf.org/public/rfc/bibxml/"
 Base URI for RFC referennces included using the "include" processing instruction.

3.3.26. rfc-uri - URI Template for HTML Version of RFCs

PI target rfc-ext
 PI pseudo-attribute rfc-uri
 XSLT parameter name xml2rfc-ext-rfc-uri
 default "https://www.rfc-editor.org/rfc/rfc{rfc}.html" (previously "https://tools.ietf.org/html/rfc{rfc}")
 URI Template for HTML Version of RFCs.

3.3.27. sec-no-trailing-dots - Disable Trailing Dots in Section Numbers

PI target rfc-ext
 PI pseudo-attribute sec-no-trailing-dots
 XSLT parameter name xml2rfc-ext-sec-no-trailing-dots
 default none
 When set to "no", do not add trailing dots to section numbers (this was the preference in the distant past).

3.3.28. std-uri - URI Template for IETF Standards

PI target rfc-ext
 PI pseudo-attribute std-uri
 XSLT parameter name xml2rfc-ext-std-uri
 default "https://www.rfc-editor.org/info/std{std}"
 URI Template for IETF Standards.

3.3.29. support-highwire-press-tags - Support 'Highwire Press' Metadata

PI target rfc-ext

PI pseudo-attribute	support-highwire-press-tags
XSLT parameter name	xml2rfc-ext-support-highwire-press-tags
default	"yes"

Decides whether the HTML transformation should generate 'Highwire Press' META tags, as used by Google Scholar.

3.3.30. support-open-graph-tags - Support 'Highwire Press' Metadata

PI target	rfc-ext
PI pseudo-attribute	support-open-graph-tags
XSLT parameter name	xml2rfc-ext-support-open-graph-tags
default	"yes"

Decides whether the HTML transformation should generate 'Open Graph' META tags, as used by Twitter.

3.3.31. support-rfc2731 - Support RFC 2731 Metadata

PI target	rfc-ext
PI pseudo-attribute	support-rfc2731
XSLT parameter name	xml2rfc-ext-support-rfc2731
default	"yes"

Decides whether the HTML transformation should generate META tags according [Section 6.4](#).

3.3.32. ucd-file - Filename of Unicode Database

PI target	rfc-ext
PI pseudo-attribute	ucd-file
XSLT parameter name	xml2rfc-ext-ucd-file
default	none

Specifies an external resource containing Unicode character database information, as described in [Section 11.24](#).

3.3.33. xml2rfc-backend - Select xml2rfc Target Version

PI target	rfc-ext
PI pseudo-attribute	xml2rfc-backend
XSLT parameter name	xml2rfc-ext-xml2rfc-backend
default	based on document date

Used in `clean-for-DTD.xslt` (see [Section 13.4](#)).

3.3.34. xref-with-text-generate - Default Handling of <xref> with Text Content

PI target	rfc-ext
PI pseudo-attribute	xref-with-text-generate
XSLT parameter name	xml2rfc-ext-xref-with-text-generate
default	"text"

Determines whether `<xref>` with text content generates additional text as in traditional text output ("text"), or just generates a link around the text ("nothing"). Note that the default might change in the future in order to achieve compatibility with other formatters.

4. Anchors

The transformation automatically generates anchors that are supposed to be stable and predictable and that can be used to identify specific parts of the document. Anchors are generated both in HTML and XSL-FO content (but the latter will only be used for PDF output when the XSL-FO engine supports producing PDF anchors).

The following anchors get auto-generated:

Anchor name	Description
rfc.abstract	Abstract
rfc.authors	Authors section
rfc.copyright	Copyright section
rfc.copyrightnotice	Copyright notice
rfc.figure. <i>n</i>	Figures (titled)
rfc.figure.u. <i>n</i>	Figures (untitled)
rfc.index	Index
rfc.ipr	Intellectual Property
rfc.iref. <i>n</i>	Internal references
rfc.note. <i>n</i>	Notes (from front section)
rfc.references	References
rfc.references. <i>n</i>	Additional references
rfc.section. <i>n</i>	Section <i>n</i>
rfc.section. <i>n</i> .p. <i>m</i>	Section <i>n</i> , paragraph <i>m</i>
rfc.status	Status of memo
rfc.table. <i>n</i>	Tables (titled)
rfc.table.u. <i>n</i>	Tables (untitled)
rfc.toc	Table of contents
rfc.xref.name. <i>n</i>	References to reference <i>n</i> to <i>name</i>

5. Supported XSLT engines

The transformation requires a non-standard extension function (see [exsl:node-set](http://www.exslt.org/exslt/functions/node-set/index.html)¹) which is however widely available. XSLT processors that do not support this extension (or a functional equivalent, such as msxsl:node-set) currently are not supported.

Input documents do not always specify the date completely. In this case, the transformation attempts to let the XSLT engine to compute the system date, using either scripting in Microsoft's XSLT engine, or the [exsl:date-time](http://www.exslt.org/exslt/functions/date-time/)² extension function.

5.1. Standalone Engines

The following XSLT engines are believed to work well:

- Windows: MSXML3 and MSXML4 (<<http://msdn.microsoft.com/xml>>; command line processor "msxsl" is available from [Microsoft Download Center](http://www.microsoft.com/download/center/)³)
- Java: Saxon (<<http://saxon.sourceforge.net/>>)
- Java: Xalan (<<http://xml.apache.org/xalan-j/>>)
- C/C++: xsltproc (libxslt) (<<http://xmlsoft.org/XSLT/>>, make sure that you have a current version, also: does not support inclusion of "https" resources -- <https://bugzilla.gnome.org/show_bug.cgi?id=791491>)

5.2. In-Browser Engines

The following browsers seem to work fine:

- Internet Explorer 6 and newer
- Firefox 3.0 and newer
 - Be aware that XSLT execution can be suppressed using [NoScript](http://www.noscript.com/)⁴
 - There seems to be a new problem in Firefox 4 where it occasionally does the initial rendering with the wrong width (people who can reproduce this problem please comment on <https://bugzilla.mozilla.org/show_bug.cgi?id=640390>.
 - Date computation is available in Firefox starting with Firefox 6 (see <https://bugzilla.mozilla.org/show_bug.cgi?id=603159>)
- Safari 3 (starting with version 3.0.4)
 - Date computation not available (see <https://bugs.webkit.org/show_bug.cgi?id=4079>)
- Google Chrome
 - Date computation not available (see <<https://bugs.chromium.org/p/chromium/issues/detail?id=1006210>>)

Note that browsers in general do not load external DTDs nor external entities (see, for instance, [Mozilla Bug 22942](https://bugzilla.mozilla.org/show_bug.cgi?id=22942)⁵) thus entities like need to be declared in the internal subset ([Appendix C.1](#)).

The following browsers are known not to work properly:

- Firefox 1.* / 2.*: (missing extension function - see change request at Mozilla BugZilla [193678](https://bugzilla.mozilla.org/show_bug.cgi?id=193678)⁶)
- Opera 9.21: execution fails, potentially to a somewhat complex XPath expression (reported to Opera as bug 245725).
- Opera 9.5 and 9.6: transformation appears to work, but CSS isn't getting applied (reported to Opera as bug 337388 on 2008-06-12).
- Safari 2.* supports client-side XSLT as of MacOS X 10.4, but misses required extension functions. A problem with stylesheets producing non-ASCII output (such as NBSP characters) has been fixed as of OSX

¹ <http://www.exslt.org/exslt/functions/node-set/index.html>

² <http://www.exslt.org/date/functions/date-time/>

³ <http://www.microsoft.com/downloads/details.aspx?FamilyID=2FB55371-C94E-4373-B0E9-DB4816552E41>

⁴ <https://addons.mozilla.org/de/firefox/addon/722>

⁵ https://bugzilla.mozilla.org/show_bug.cgi?id=22942

⁶ http://bugzilla.mozilla.org/show_bug.cgi?id=193678

10.4.4. Both problems have been reported through Apple's bug tracking system, see <<http://drakken.dbc.mtview.ca.us/pipermail/xml2rfc/2005-May/002073.html>> and <http://bugs.webkit.org/show_bug.cgi?id=4079>.

6. Transforming to HTML

Transformation to HTML can be done inside the browser if it supports XSLT. To enable this, add the following processing instruction to the start of the source file:

```
<?xml-stylesheet type='text/xsl' href='rfc2629.xslt' ?>
```

(and ensure that `rfc2629.xslt` is present).

6.1. HTML compliance

The transformation result is supposed to conform to HTML5 [HTML5]. This can be checked using the W3C's online validator at <http://validator.w3.org>.

XSLT 1.0 is not capable to directly emit the HTML doctype declaration, thus uses the SYSTEM ID "about:legacy-compat" instead (see [Section 8.1.1](#) of [HTML5]).

When not run in a browser, the doctype declaration can be adjusted using a small script, such as with:

```
saxon test.xml rfc2629.xslt | awk -f html5doctype.awk
```

with

```
#!/usr/bin/awk -f

# waitfordoctype:
# 0: wait for line starting with DOCTYPE and eat empty lines
# 1: wait for line starting with <html
# 2: afterwards

BEGIN {
    waitfordoctype = 0;
}

/<!DOCTYPE .*/ {
    if (waitfordoctype == 0) {
        waitfordoctype = 1
    }
}

/<html.*/ {
    if (waitfordoctype == 1) {
        waitfordoctype = 2
        printf("<!DOCTYPE html>\n")
    }
    else {
        print
    }
}

{
    if (waitfordoctype == 0 && $0 != "") {
        print
    }
    else if (waitfordoctype == 2) {
        print
    }
}
```

6.2. Standard HTML LINK elements

LINK elements exist since HTML 2.0. They can be used to embed content-independent links inside the document. Unfortunately, only few user agents support this element. Firefox users may want to check the [Link Widgets](https://addons.mozilla.org/firefox/2933/)⁷ extension.

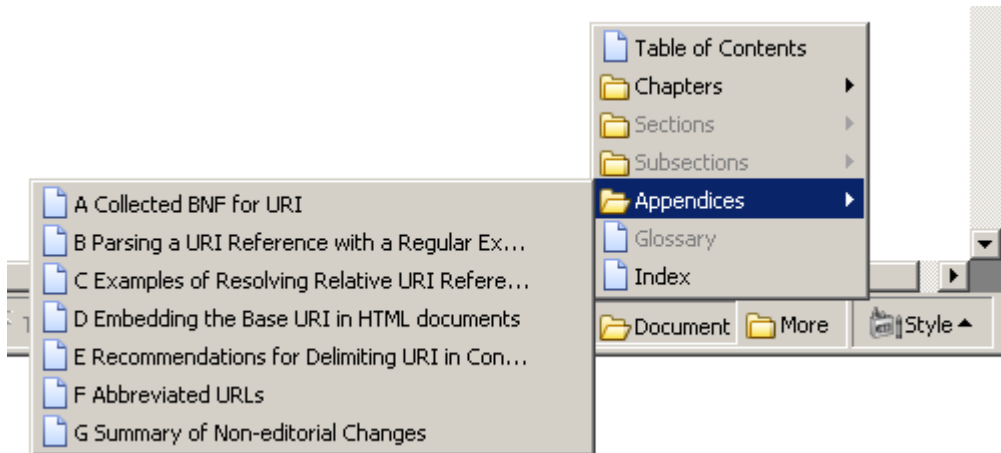
The following LINK elements are produced:

LINK type	description
alternate	for RFCs, a link to the authoritative ASCII version on the IETF web site
appendic	pointer to all top-level appendics
author	pointer to "authors" section
chapter	pointer to all top-level sections
contents	pointer to table of contents

⁷ <https://addons.mozilla.org/firefox/2933/>

LINK type	description
copyright	pointer to copyright statement
index	pointer to index

The figure below shows how Mozilla Firefox 1.0 displays the Site Navigation Bar for rfc2396.xml.



6.3. Standard HTML metadata

The following standard HTML META elements are produced:

META name	description
generator	from XSLT engine version and stylesheet version
keywords	from keyword elements in front section

6.4. Dublin Core (RFC2731) metadata

Unless turned off using the "rfc-ext support-rfc2731" processing instruction, the transformation will generate metadata according to [RFC2731] and [DC-HTML].

The following DCMI properties are produced:

META name	description
DC.Creator	from author information in front section
DC.Date.Issued	from date information in front section
DC.Description.Abstract	from abstract
DC.Identifier	document URN [RFC2648] from "docName" attribute
DC.isPartOf	RFC ISSN (for RFCs)
DC.Relation.Replaces	from "obsoletes" attribute

6.5. Insertion of Live RFC Meta Data

RFCs are immutable; once published, they do not change anymore. What *does* change though is their status, their relation to subsequent RFCs (such as when they are updated), and errata.

`rfc2629toXHTML.xslt` by default inserts code that will pull the relevant information from <https://tools.ietf.org>. This can be disabled by specifying the parameter "xml2rfc-ext-insert-metadata=no" (or by inserting the equivalent processing instruction into the source code).

An example for the generated information can be seen at <https://greenbytes.de/tech/webdav/rfc2616.html#rfc.meta>.

Note: the status information really should be available from the RFC Editor, right now it only exists only in HTML form (such as in <<https://www.rfc-editor.org/info/rfc2616>>. Furthermore, the service provided by <<https://tools.ietf.org>> is considered experimental, so this feature might be changed or removed without prior notice.

...and no, we currently can't obtain the exact list of errata, just a flag indicating whether errata exist.

6.6. Insertion of RFC Errata Links

Unfortunately, the RFC Editor does not provide errata information in a well-defined machine readable format. What's available is "regular" HTML (and that could be the worst currently in use in standards bodies...).

parse-errata.xslt attempts to parse useful information out of these pages.

It can be run like that (requires an XSLT2 processor):

```
# get the raw html and strip form feed characters
curl -s https://www.rfc-editor.org/errata_search.php?rfc=2616 \
| tr -d '\f' > rfc2616.rawerrata
# regexps are your friend
saxon97he parse-errata.xslt parse-errata.xslt doc=2616 > rfc2616.errata
```

The code tries to make sense of the HTML, in particular it tries to detect what RFC sections each erratum applies to. The resulting XML format is work-in-progress and just contains the information that will be useful in subsequent formatting of the RFC.

When *formatting* the RFC for HTML output, the errata file can be passed as stylesheet parameter ("xml2rfc-ext-errata"). The output will include errata links at the beginnings of the section they apply to, or at the beginning of Section 1 when the location is unknown.

For the sake of embedding, three types of errata are relevant; their type is indicated with a symbol:

- "held for update": reviewed and decided to be "held for document update" (these are mostly editorial); "scissors" symbol.
- "submitted": submitted, but not reviewed; "letter" symbol.
- "verified": reviewed and found to be correct; "checkmark" symbol.

To recap: the errata information is passed into the transformation as additional parameter. The errata information will *not* be automatically retrieved from the RFC Editor web site.

Finally, here's an example for inserted errata links: <<https://greenbytes.de/tech/webdav/rfc7230.html#transfer.codings>>.

7. Transforming to XHTML

Transforming to XHTML requires slightly different XSLT output options and is implemented by the derived transformation script `rfc2629toXHTML.xslt`.

8. Transforming to CHM (Microsoft Compiled Help)

To generate a CHM file using Microsoft's HTML Help Compiler (hhc), three files are required in addition to the HTML file.

1. hhc - table of contents file (HTML)
2. hhk - index file (HTML)
3. hhp - project file (plain text)

The three files are generated with three specific transformations, each requiring the additional XSLT parameter "basename" to specify the filename prefix.

Example:

```
saxon rfc2616.xml rfc2629toHhp.xslt basename=rfc2616 > rfc2616.hhp
saxon rfc2616.xml rfc2629toHhc.xslt basename=rfc2616 > rfc2616.hhc
saxon rfc2616.xml rfc2629toHhk.xslt basename=rfc2616 > rfc2616.hhk
hhc rfc2616.hhp
```

9. Transforming to PDF

9.1. Via XSL-FO

Transformation to XSL-FO [XSL-FO] format is available through `rfc2629toFO.xslt` (which includes `rfc2629.xslt`, so keep both in the same folder).

Compared to HTML user agents, XSL-FO engines unfortunately either come as open source (for instance, Apache FOP) or feature-complete (for instance, AntennaHouse XSL Formatter), but not both at the same time.

As Apache FOP needs special workarounds (index generation), and some popular extensions aren't standardized yet, the translation produces a generic output (hopefully) conforming to [XSL-FO]. Specific backends (`xsl111toFop.xslt`, `xsl111toXep.xslt`, `xsl111toAn.xslt`) then provide post-processing for the individual processors.

Note: the output is currently targeted at Apache FOP 1.1.

9.1.1. Example: producing output for Apache FOP

Example:

```
saxon rfc2616.xml rfc2629toFo.xslt > tmp.fo
saxon tmp.fo xsl111toFop.xslt > rfc2629.fo
```

9.2. Via HTML

PDF output can also be produced directly from (X)HTML. One simple approach is to rely on the browser's printing function, and to use a printer driver that produces PDF. Depending on the browser's CSS capabilities, the output will behave properly with respect to table breaks etc.

Alternatives are tools which can produce PDF directly from (X)HTML input, based on the CSS printing information. For instance: "PrinceXML" (<<http://www.princexml.com/>>) and "WeasyPrint" (<<https://weasyprint.org/>>).

For instance, PDF output can be produced with:

```
saxon input.xml rfc2629.xslt > output.html
weasyprint output.html output.pdf
```

10. Transforming to ePub

Experimental transformation to ePub format is available through a set of stylesheets, and the Unix Shell script `mkepub.sh` (which requires that "zip" and either "saxon" or "xsltproc" are installed).

For instance, an epub version of `rfc2616.xml` can be generated like this:

```
mkepub.sh rfc2616.xml
```

11. Generic Extensions

This section documents extensions implemented in `rfc2629.xslt`, using the extension namespace "`http://purl.org/net/xml2rfc/ext`".

11.1. `<abnf-char-sequence>` element

Converts the contained quoted string into a hex-encoded character sequence, for use in case-sensitive ABNF productions.

For instance, "`<x:abnf-char-sequence>\"HTTP\"</x:abnf-char-sequence>`" gets converted to "`%x48.54.54.50`".

11.2. `<anchor-alias>` element

Using its "value" attribute, this element allows the definition of an internal link target alias for the enclosing element. This alias can then be used with the `<ref>` element for intra-document references.

Note that the anchor alias is not subject to the naming constraints that apply to anchor elements (which are [XML names](#)⁸).

11.3. `<bb>` element

Marking up a string as `<bb>` indicates that it represents the bottom line of a box drawing, replacing the "+" and "-" characters accordingly.

11.4. `<bc>` element

Marking up a string as `<bc>` indicates that it represents a center line of a box drawing, replacing the "|" character accordingly.

11.5. `<bcp14>` element

This element marks the content as being one of the normative keywords defined in [\[RFC2119\]](#).

⁸ <http://www.w3.org/TR/REC-xml/#NT-Name>

The DOCTYPE definition below allows using these keywords using XML entity expansion: such as in "...server &MUST; accept...".

```
<!DOCTYPE rfc [
  <!ENTITY MAY "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >MAY</bcp14>">
  <!ENTITY MUST "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >MUST</bcp14>">
  <!ENTITY MUST-NOT "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >MUST NOT</bcp14>">
  <!ENTITY OPTIONAL "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >OPTIONAL</bcp14>">
  <!ENTITY RECOMMENDED "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >RECOMMENDED</bcp14>">
  <!ENTITY REQUIRED "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >REQUIRED</bcp14>">
  <!ENTITY SHALL "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >SHALL</bcp14>">
  <!ENTITY SHALL-NOT "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >SHALL NOT</bcp14>">
  <!ENTITY SHOULD "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >SHOULD</bcp14>">
  <!ENTITY SHOULD-NOT "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'
    >SHOULD NOT</bcp14>"> ]>
```

11.6. <blockquote> element

This element is like the <blockquote> element defined in [Section 4.4.4](#) of [HTML5] (note this is a block-level element!). It should contain one or more <t> child elements.

11.7. <bt> element

Marking up a string as <bt> indicates that it represents the top line of a box drawing, replacing the "+" and "-" characters accordingly.

11.8. <c> element

Contains mapping information for a single Unicode code points. Attributes are "c" (the actual character), "n" (the code point), and "d" (the name/description).

For instance:

```
<x:u-map>
  <x:c n="8364" c="&#8364;" d="EURO SIGN"/>
<x:u-map>
```

(The format is deliberately terse so that the size of a mapping file containing the whole Unicode character database is minimized).

11.9. <contributor> element

This element can be used to include a contributor's contact information in place where a paragraph (<t>) would be allowed otherwise.

See [Section 12.8](#) for a standard alternative.

11.10. <dfn> element

This element is like the <dfn> element defined in [Section 4.5.8](#) of [HTML5].

11.11. <feedback> element

This elements allows declaring a feedback link for document reviewers. The template string takes the form of a URI template, such as:

```
<x:feedback template="mailto:ietf-http-wg@w3.org?subject={docname} ,
%20%22{section}%22&amp;body=&lt;{ref}&gt; : " />
```

where "docname" is substituted by the document name, "section" is substituted by section title (number and name), and "ref" is substituted by a URI pointing to the section being referenced.

11.12. <h> element

This element is like the "[h](#)" element in [XHTML2].

11.13. <highlight> element

Used to highlight text passages, currently only allowed in <artwork>.

Note: this is stripped when generating input for xml2rfc, so please use with care.

11.14. <length-of> element

This element can be used to insert the length of another formatted section (in decimal).

Example: computing the Content-Length header value

```
<artwork>
...
Content-Length: <x:length-of target="req" />

<x:span anchor="req">123456789
<x:span><artwork/>
```

The lenght computation counts line ends as two characters (CRLF).

Note that indentation characters in artwork *will* be counted. The "indented" attribute allows to specify the amount of indentation to be subtracted from the computed length.

11.15. <link> element

This element can be added as a top-level child element below <rfc> to indicate additional link information. It's currently used only when generating HTML output, in which case an HTML [<link>](#)¹⁰ element with identical attributes gets generated.

⁹ http://www.w3.org/TR/2006/WD-xhtml2-20060726/mod-structural.html#edef_structural_h

¹⁰ <http://www.w3.org/TR/html4/struct/links.html#edef-LINK>

Example: generating HTML link element

```
<x:link xmlns:x="http://purl.org/net/xml2rfc/ext"
  rel="Bookmark"
  title="IETF WEBDAV Working Group"
  href="http://ftp.ics.uci.edu/pub/ietf/webdav/" />
```

If the attribute "basename" is present, it is used to compute the target href based on the output format being generated (this is handy for "next"/"prev" links in a series of documents. In this case, the href attribute is not required.

For instance:

```
<x:link xmlns:x="http://purl.org/net/xml2rfc/ext"
  rel="next"
  title="Part2"
  basename="draft-foobar-protocol-p2-latest" />
```

11.16. <lt> element

Used for grouping multiple <t> elements into a single list item.

11.17. <note> element

Can be used to add a note, usually indented by a few characters. It should contain one or more <t> child elements.

11.18. <parse-xml> element

This element instructs the processor to parse the contents as XML and to warn when there's a problem (requires either MSXML or Saxon8 or newer).

11.19. <prose> element

This element can be used inside <reference> to add plain text (before the date, when present).

See also <refcontent> ([Section 12.21](#)).

11.20. <q> element

This element is like the <q> element defined in [Section 4.5.7](#) of [HTML5].

11.21. <ref> element

This element is a simplified variant of the <xref> element, in that no "target" attribute needs to be specified, instead the text contents acts as identifier. That in itself wouldn't be terribly useful, but together with the <anchor-alias>, it allows referring to other parts of the document with minimal additional markup.

For instance, given an alias definition such as


```
<section title="Test" anchor="test">
  <x:anchor-alias value="alias1"/>
  <x:anchor-alias value="alias 2"/>
  ...
</section>
```

the following simple references

```
<x:ref>test</x:ref>
<x:ref>alias1</x:ref>
<x:ref>alias 2</x:ref>
```

are equivalent to...:

```
<xref target="test" format="none">test</xref>
<xref target="test" format="none">alias1</xref>
<xref target="test" format="none">alias 2</xref>
```

11.22. <source> element

Can be used to enhance a <reference> with information about the location for the XML source. This can be used by the <xref> processing code to automatically extract the target section number (see [Section 11.32](#)).

For example:

```
...
<xref target="RFC2616" x:fmt="of" x:rel="#PUT" />
...

<reference target="RFC2616"/>
  ...
  <x:source href="rfc2616.xml"/>
  ...
```

11.23. <sup> element

This element is like the <sup> element in [Section 4.5.16](#) of [HTML5].

Note: the down conversion to RFC7749 format replaces "x^y" by "x^y".

See also <sup> ([Section 12.30](#)).

11.24. <u-map> element

Needed for the lookup of Unicode character database information; this element can either appear in-band in the source document, or off-band as specified using the xml2rfc-ext-ucd-file parameter/PI.

Contains multiple <x:c> elements (see [Section 11.8](#)).

The utility XSLT `convert-ucd.xslt` can be used to create a mapping file based on the Unicode XML database format, as available from <https://unicode.org/ucd/#UCDinXML>.

11.25. Extensions to <artwork> element

Sometimes, artwork occurs inside lists. To get it indented properly in xml2rfc's text output, it needs to be indented in the source. This is sub-optimal, as this whitespace will also appear in the HTML output, where it's already indented due to HTML's semantics. As a workaround, a "x:indent-with" attribute can be specified, containing a string that will be prepended to each line when `clean-for-DTD.xslt` is run (see [Section 13.4](#)).

Furthermore, documents can contain code that might need to be marked as "code component" (<http://www.ietf.org/iesg/statement/copyright.html>). This can be done using "x:is-code-component".

Finally, when allowing pretty-printing of code (see "html-pretty-print" in [Section 3.3](#), the "x:lang" attribute can be used to explicitly opt into pretty-printing. If the pretty printer can figure out the code type without assistance, an empty value will be sufficient. Otherwise, the language can be specified in the attribute (such as "html"), which will in turn be coded into the CSS class as "lang-" concatenated with the language name.

11.26. Extensions to <date> element

The attribute "x:include-day" ("true", "false") can be used to control whether the day-of-month should be included in the output. This can be used to adjust the rendering of dates for April-1st-RFCs to the desired special format, including the "1".

In particular: on the front page, the attribute defaults to "true", *unless* when generating RFCs. The output format also varies depending on RFC-ness: it's "day month year" for RFCs, but "month day, year" otherwise. In references, the attribute defaults to "false", and when set to true, the day is just inserted in front of the month.

Furthermore, this implementation supports the extension discussed in in [Section 4.1.3.2](#) of [V3IMPNOTES]: in absence of any attributes, a "vague" date can be given as text content (this also applies to the case where <date> appears below a <reference> element).

11.27. Extensions to <eref> element

For v3 documents, angle brackets will only be produced if the attribute "brackets" is set to "angle".

11.28. Extensions to <iref> element

The extension attribute below is allowed on the standard <iref> element:

- x:for-anchor specifies that the <iref> will also be automatically inserted whenever the specified anchor is cross-referenced -- this may save entering lots of <iref> instances. As a special case, a value of "" (empty string) refers to the anchor attribute of the closest ancestor.

11.29. Extensions to <list> element

The extension attribute below is allowed on the standard <list> element:

- x:indent specifies the amount of indentation for list items in hanging lists. This can be useful when the output format, such as XSL-FO, does not support automatical formatting. The value takes an XSL-FO width, such as "5em". The default is *length of longest label in characters times 0.8em*.

Also, the <list> element can take <x:lt> child elements instead of <t>, allowing to insert multiple paragraphs into a single list item.

11.30. Extensions to <rfc> element

The extension attributes below are allowed on the standard <rfc> element:

- grddl:transformation can be used to reference a GRDDL transform.
- x:maturity-level can be used to specify the IETF Standards Track Maturity Level of "proposed", "draft" or "internet" (see [Section 4.1](#) of [RFC2026]).

11.31. Extensions to <section> element

The extension attribute below is allowed on the standard <list> element:

- x:fixed-section-number can be used to specify a fixed section number. This can be useful when formatting historic documents that used a different numbering style.

11.32. Extensions to <xref> element

Four extension attributes are allowed on the standard <xref> element:

1. x:sec can be specified to point to a specific section of the referenced document,
2. x:rel may specify a relative reference to use when linking into the referenced document (if linking by section number is not available),
3. x:fmt defines the text format to be used,
4. x:title provided a title for the link (to be used in the HTML title attribute).

The following formats are defined for the x:fmt attribute:

, (Comma)	[<i>reference</i>], Section <i>sec</i>
()	[<i>reference</i>] (Section <i>sec</i>)
of	Section <i>sec</i> of [<i>reference</i>]
number	<i>sec</i>
none	No output (can be used to have xrefs to references without having them rendered as such)
sec	Section <i>sec</i>

These extensions are currently only supported for <xref> elements without child nodes.

If the processor knows how to reference the target section, it will generate a link directly to the target section, such as in [\[RFC2119\]](#), [Section 5](#).

12. Experimental Support for RFC7991bis Vocabulary

`rfc2629.xslt` experimentally supports some elements from the "V3" vocabulary, defined in [\[RFC7991bis\]](#).
This support is experimental, as the "v3" vocabulary is still being developed.

12.1. artset Element

See [Section 3.1.1](#) of [\[V3IMPNOTES\]](#).

12.2. aside Element

See [Section 2.7](#) of [\[RFC7991bis\]](#).

12.3. bcp14 Element

See [Section 2.10](#) of [\[RFC7991bis\]](#).

12.4. blockquote Element

See [Section 2.11](#) of [\[RFC7991bis\]](#).

12.5. boilerplate Element

See [Section 2.11](#) of [\[RFC7991bis\]](#).

12.6. br Element

12.7. cityarea Element

See [Section 4.2.7](#) of [\[V3IMPNOTES\]](#).

12.8. contact Element

Can be used to embed contact information in a layout *similar* to the one used for authors; mainly for use in a "Contributors" section.

When used in phrase content, only the name or the organization (absent a name) will be displayed.

12.9. displayreference Element

See [Section 2.20](#) of [\[RFC7991bis\]](#).

12.10. dd Element

See [Section 2.19](#) of [\[RFC7991bis\]](#).

12.11. dl Element

See [Section 2.21](#) of [\[RFC7991bis\]](#).

12.12. dt Element

See [Section 2.22](#) of [RFC7991bis].

12.13. em Element

See [Section 2.23](#) of [RFC7991bis].

12.14. extaddr Element

See [Section 4.2.7](#) of [V3IMPNOTES].

12.15. li Element

See [Section 2.31](#) of [RFC7991bis].

12.16. link Element

See [Section 2.32](#) of [RFC7991bis].

12.17. name Element

See [Section 2.34](#) of [RFC7991bis]. *Currently only supported inside <references> and <section>.*

12.18. ol Element

See [Section 2.36](#) of [RFC7991bis].

12.19. pobox Element

See [Section 4.2.7](#) of [V3IMPNOTES].

12.20. postalLine Element

See [Section 2.41](#) of [RFC7991bis].

12.21. refcontent Element

See [Section 2.42](#) of [RFC7991bis].

12.22. Extensions to reference Element

See [Section 2.43](#) of [RFC7991bis].

12.22.1. quoteTitle attribute

See [Section 2.39.2](#) of [RFC7991bis-01].

12.23. referencegroup Element

See [Section 2.44](#) of [RFC7991bis].

12.24. relref Element

See [Section 2.47](#) of [RFC7991bis].

EXPERIMENTAL, might be removed again - see <<https://github.com/rfc-format/draft-iab-xml2rfc-v3-bis/issues/26>>.

12.25. Extensions to section Element

See [Section 2.49](#) of [RFC7991bis].

12.25.1. numbered attribute

See [Section 2.49.2](#) of [RFC7991bis].

12.25.2. removeInRFC attribute

See [Section 2.49.3](#) of [RFC7991bis].

12.26. sortingcode Element

See [Section 4.2.7](#) of [V3IMPNOTES].

12.27. sourcecode Element

See [Section 2.52](#) of [RFC7991bis].

12.28. strong Element

See [Section 2.54](#) of [RFC7991bis].

12.29. sub Element

See [Section 2.55](#) of [RFC7991bis].

12.30. sup Element

See [Section 2.56](#) of [RFC7991bis].

12.31. svg Element

See [Section 4](#) of [RFC7991bis].

12.32. table Element

See [Section 2.58](#) of [RFC7991bis].

12.33. tbody Element

See [Section 2.59](#) of [RFC7991bis].

12.34. td Element

See [Section 2.60](#) of [RFC7991bis].

12.35. tfoot Element

See [Section 2.61](#) of [RFC7991bis].

12.36. th Element

See [Section 2.62](#) of [RFC7991bis].

12.37. thead Element

See [Section 2.63](#) of [RFC7991bis].

12.38. tr Element

See [Section 2.65](#) of [RFC7991bis].

12.39. tt Element

See [Section 2.66](#) of [RFC7991bis].

12.40. u Element

See [Appendix A.1](#) of [V3IMPNOTES].

Note that in this implementation, this element needs mapping information in two cases:

1. When in XSLT 1.0: to compute the Unicode code point of a given character (when non-ASCII),
2. In any case: to lookup the Unicode name of a character.

The mapping information can be supplied inline using the `<x:u-map>` element, or in an external file. See [Section 11.24](#) for details.

12.41. ul Element

See [Section 2.67](#) of [RFC7991bis].

12.42. Extensions to xref Element

12.42.1. relative attribute

See [Section 11.32](#), "x:rel" attribute.

12.42.2. section attribute

See [Section 11.32](#), "x:sec" attribute.

12.42.3. sectionFormat attribute

See [Section 11.32](#), "x:fmt" attribute, where "bare" maps to "sec", "comma" maps to ",", "of" maps to "of", and "parens" maps to "()".

12.43. xi:include Element

See [Appendix B.1](#) of [RFC7991bis].

EXPERIMENTAL and INCOMPLETE - only supported as a child element of <references>, and only supporting parse type "XML".

13. Utilities

13.1. Checking References

`check-references.xslt` can be used to check all references to RFC- and ID-series IETF publications and to W3C publications (note this script requires local copies of <http://ftp.isi.edu/in-notes/rfc-index.xml> and <http://www.w3.org/2002/01/tr-automation/tr.rdf> and will use the XML status information provided at <http://tools.ietf.org/>).

If the document is supposed to be published on the IETF standards track, the desired level can be specified using the parameter `intended-level` as 'proposed', 'draft' or 'internet'. Alternatively, it can be specified inside the document using the attribute `x:maturity-level` on the `<rfc>` element (see [Section 11.30](#)).

Note: Downward references should be annotated using the `<annotate>` element, containing an `<xref>` to [\[BCP97\]](#).

When an XSLT 2.0 processor is used, links in the document can be checked as well using the `link-check` parameter ('yes' or 'no'). Note that this only works for http links to documents of type text/*.

For instance, as of 2008-07-12, the script produces for <https://greenbytes.de/tech/webdav/rfc2518.xml>:

```
> saxon rfc2518.xml check-references.xslt intended-status=PROPOSED \
  link-check=yes
```

Normative References:

```
ISO-11578: not checked
ISO-639: not checked
ISO-8601: not checked
REC-xml-19980210: [FirstEdition] obsoleted by REC-xml-20001006
REC-xml-names-19990114: [FirstEdition] obsoleted by
  REC-xml-names-20060816
RFC1766: [PROPOSED STANDARD] obsoleted by RFC3066 RFC3282
RFC2068: [PROPOSED STANDARD] obsoleted by RFC2616
RFC2069: [PROPOSED STANDARD] obsoleted by RFC2617
RFC2119: [BEST CURRENT PRACTICE] (-> BCP0014) ok
RFC2141: [PROPOSED STANDARD] ok
RFC2277: [BEST CURRENT PRACTICE] (-> BCP0018) ok
RFC2396: [DRAFT STANDARD] obsoleted by RFC3986
RFC2279: [DRAFT STANDARD] obsoleted by RFC3629
```

Informational References:

```
REC-PICS-labels-961031: [REC] ok
RFC1807: [INFORMATIONAL] ok
RFC2026: [BEST CURRENT PRACTICE] (-> BCP0009) ok
RFC2291: [INFORMATIONAL] ok
RFC2376: [INFORMATIONAL] obsoleted by RFC3023
RFC2413: [INFORMATIONAL] obsoleted by RFC5013
USMARC: not checked
WF: not checked
```

Link Targets

```
<http://www.w3.org/TR/1998/REC-xml-19980210>: ok
<http://www.w3.org/TR/1999/REC-xml-names-19990114>: ok
<http://www.dlib.org/dlib/july96/lagoze/07lagoze.html>: ok
<http://www.w3.org/pub/WWW/TR/REC-PICS-labels-961031.html>: ok
```


Recognized formats in the <seriesInfo> element are:

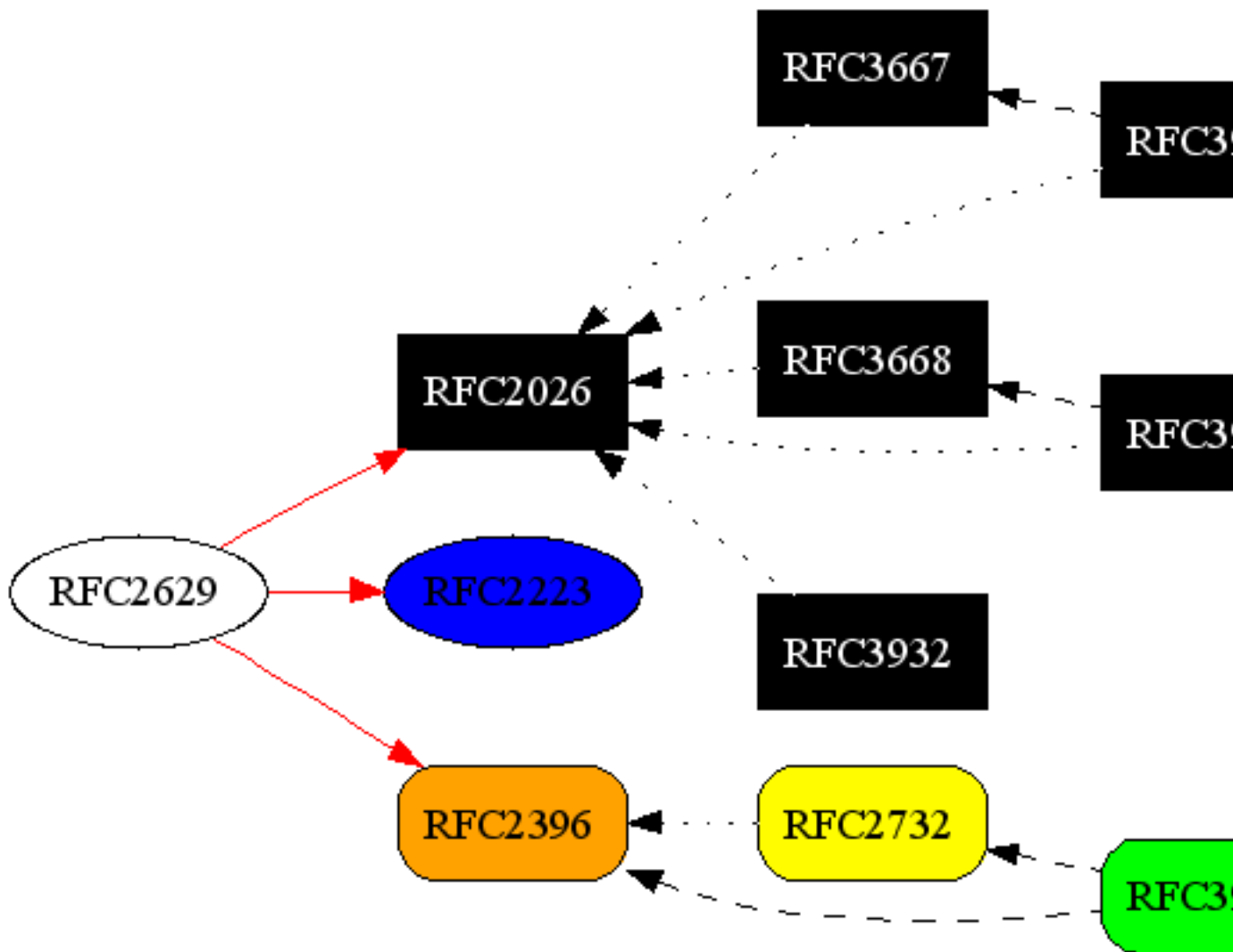
- for RFCs, the name attribute must be "RFC", and the value attribute must be the number of the RFC,
- for Internet Drafts, the name attribute must be "ID" or "Internet-Draft", and the value attribute must be the file name of the draft (including the two-digit running number, but excluding a file extension),
- for W3C documents, the name attribute must be "W3C", must start with or must start with "World Wide Web and the value attribute must be the "shorthand" name of the specification, such as "REC-xml-19980210".

Note: this stylesheet will need network access to check links and status of Internet Drafts. When running a Java-based XSLT engine, you may have to supply Java system properties specifying the HTTP proxy to be used, such as "-Dhttp.proxyHost=hostname -Dhttp.proxyPort=80".

13.2. Generating Graphs from References

`gen-reference-graph.xslt` generates a graph of RFC dependencies, using the same base data as in `check-references.xslt` (see [Section 13.1](#)). Its output is a "dot" file, to be processed by GraphViz (see <http://www.graphviz.org/>).

The picture below shows the RFC dependencies in RFC2629.



13.3. Producing reference entries for books

`amazon-asin.xslt` uses the Amazon web services to generate a `<reference>` element for a given ASIN (ISBN).

For instance:

```
<?xml version="1.0" encoding="utf-8"?>
<references>
  <reference target="urn:isbn:0134516591">
    <front>
      <title>Simple Book, The: An Introduction to Internet Management,
        Revised Second Edition</title>
      <author surname="Rose"
        fullname="Marshall T. Rose" initials="M. T. ">
        <organization/>
      </author>
      <author surname="Marshall"
        fullname="Rose T. Marshall" initials="R. T.">
        <organization/>
      </author>
      <date year="1996" month="March"/>
    </front>
    <seriesInfo name="Prentice Hall" value=""/>
  </reference>
</references>
```

Note that the resulting XML usually requires checking, in this case Amazon's database is playing tricks with Marshall's name...

13.4. Down-converting to RFC 7749 Grammar

`clean-for-DTD.xslt` can be used to down-convert some extensions to a format that is supported by the existing `xml2rfc` formatters, mainly for the purpose of generating plain-text output. Note that these extensions are experimental (feedback appreciated).

The following mappings are done:

- `<iref>` elements inside `<artwork>` elements are moved in front of the enclosing `<figure>` element.
- `<xref>` elements inside `<artwork>` are expanded just like in regular text (that is, the markup is stripped, but the element is replaced by the applicable replacement text).
- `<x:anchor-alias>` elements get stripped.
- `<x:bcp14>` elements get stripped.
- `<x:bb>`, `<x:bc>` and `<x:bt>` elements get stripped.
- `<x:blockquote>` elements get converted to indented text (through a `<list>` element).
- `<x:dfn>` elements get stripped.
- `<x:h>` elements get stripped.
- `<x:link>` elements get stripped.
- `<x:lt>` elements get collapsed into a single `<lt>` element with added `<vspace>` added to simulate paragraph breaks.
- `<x:note>` elements get converted to indented text (through a `<list>` element).
- `<x:q>` elements get stripped, with apostrophes added around the text.
- `<x:prose>` elements are transformed into `<seriesInfo>` elements (which is an abuse of the element and only a workaround until `xml2rfc` gets a matching extension).
- `<x:ref>` elements get replaced by `<xref>` elements, targetting either the anchor or another anchor with matching `<x:anchor-alias>` child element.

Note: the above list is known to be incomplete and needs work. For instance, most of the extensions described in [Section 12](#) get some mapping as well.

As the output formatters evolve to support the V3 format (proposed in [\[RFC7991bis\]](#)), `clean-for-DTD.xslt` will start taking advantage of these changes. Right now, it supports three modes, one of which being used for the historic TCL processor, and the other ones being used with `xml2rfc 2.5.2` and `xml2rfc 2.6.0` (see <https://pypi.python.org/pypi/xml2rfc/>¹¹).

The modes can be selected using the `xml2rfc-ext-xml2rfc-backend` parameter or the `rfc-ext/xml2rfc-backend` processing instruction. The default mode is "201610" for documents with a publication date between January and May 2017, "201706" for documents newer than May 2017, and "201510" otherwise:

- 201510** For `xml2rfc.tcl` and versions of the Python formatter before version 2.5.2.
- 201610** Maps the V3 extension for unnumbered section numbers to something the 2.5.2 version of the Python formatter understands (but note <https://trac.tools.ietf.org/tools/xml2rfc/trac/ticket/313>¹²).
- 201706** Copies the V3 extension for unnumbered section numbers to the output.

13.5. Extracting artwork

With `extract-artwork.xslt`, artwork elements named through the "name" attribute can be extracted. This can be used to automatically check their syntax (for instance, when ABNFs appear within a figure element).

For instance:

```
saxon rfc3986.xml extract-artwork.xslt name=uri.abnf
```

In addition, artwork of a specific type can be extracted, such as with:

```
saxon rfc3986.xml extract-artwork.xslt type=abnf
```

When extracting by type, artwork elements with a specified name can be excluded; this can be handy when the document uses some kind of schema language, and an appendix contains the collected schema, repeating definitions from earlier on. Example:

```
saxon rfc3986.xml extract-artwork.xslt type=abnf except-name=clschm
```

13.6. GRRDL

`rfc2629grddl.xslt` extracts RDF information. This is experimental work-in-progress. See <http://www.w3.org/TR/grddl/> for more information.

13.7. HTML Live Refresh

Experimental

The "HTML Live Refresh" mode allows to run a text editor and a browser side-by-side, with the browser auto-updating every few seconds, displaying the updated HTML, and automatically navigating to the part of the page that changed last.

The requirements for this mode are:

1. A browser that supports the `DOMParser` and `XSLTProcessor` APIs.
2. The ability to reload the source code and the XSLT code from within Javascript; in some browsers this is forbidden for "file:" URIs due to perceived security problems.

This feature is currently tested with:

¹¹ <https://pypi.python.org/pypi/xml2rfc/>

¹² <https://trac.tools.ietf.org/tools/xml2rfc/trac/ticket/313>

1. Mozilla Firefox (where, to be able to reload from the local filesystem, Firefox needs to be configured with `security.fileuri.strict_origin_policy` set to `false` in `about:config`, see [Mozilla documentation](#)¹³), and
2. Google Chrome (where, to be able to reload from the local filesystem, Chrome needs to be started with the command line option `--allow-file-access-from-files`).

Use of this feature requires the inclusion of a processing instruction that holds the name of the XML source, such as:

```
<?rfc-ext refresh-from="draft-foo-bar-00.xml"?>
```

The optional parameters `refresh-interval` and `refresh-xslt` support changing the refresh interval (in seconds, defaulting to 10), and the name of the XSLT file to use (defaulting to "rfc2629.xslt").

To test this feature, start with a minimal source file like this:

```
<?xml-stylesheet type='text/xsl' href='rfc2629.xslt'?>
<?rfc-ext refresh-from="draft-foo-bar-00.xml"?>
<rfc docName="draft-foo-bar-00" ipr="trust200902">
  <front>
    <title>Title Goes Here</title>
    <abstract>
      <t>Abstract</t>
    </abstract>
  </front>
</rfc>
```

...open it in both text editor and browser, start editing and of course occasionally save. And don't forget to put `rfc2629.xslt` into the same folder.

13.8. Refreshing included material in the XML source

There are many methods for automatic inclusion of material in the XML source, such as the "include" processing instruction (see [Section 3.1](#)), external entities ([Appendix C.1](#)), or XInclude. In general, those share a common problem: the XML source file isn't self-contained, which makes it harder to submit it as Internet Draft.

The tool `refresh-inclusions.sh` does in-place replacement: it scans the source file for inclusion directives (expressed as XML processing instructions), and refreshes the included text with data from an external file. It will not modify the source file unless included material did actually change. When it does modify the source file, it will copy the original source to a backup file.

`refresh-inclusions.sh` can include both plain text (`BEGINESCAPEDINC/ENDESCAPEDINC`) and XML (`BEGININC/ENDINC`). The figure below was inserted using:

```
<?BEGINESCAPEDINC refresh-inclusions.sh ?>
...
<?ENDSCAPEDINC refresh-inclusions.sh ?>
```

(note that the SP character at the end of the directive is significant)

¹³ https://developer.mozilla.org/en-US/docs/Archive/Misc_top_level/Same-origin_policy_for_file:_URIs

refresh-inclusions.sh:

```
#!/bin/sh
# Refresh file inclusions based on XML processing instructions
#
# Copyright (c) 2006-2023, Julian Reschke (julian.reschke@greenbytes.de)
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
#
# * Redistributions of source code must retain the above copyright notice,
#   this list of conditions and the following disclaimer.
# * Redistributions in binary form must reproduce the above copyright notice,
#   this list of conditions and the following disclaimer in the documentation
#   and/or other materials provided with the distribution.
# * Neither the name of Julian Reschke nor the names of its contributors
#   may be used to endorse or promote products derived from this software
#   without specific prior written permission.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
# AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
# IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
# ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
# LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
# CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
# SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
# INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
# CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
# POSSIBILITY OF SUCH DAMAGE.

expand() {

    # remember whether we started with CRLF (assumes that we have dos2unix)
    CRLF=$(dos2unix -ic "$1" 2>/dev/null | tr -d ' ')

    cat "$1" | tr -d "\015" | awk '

function filecontents(filename) {
    while (getline < filename > 0) {
        fc[filename] = fc[filename] $0 "\n"
    }
    return fc[filename]
}

BEGIN {
    includefile = "";
    includeescapedfile = "";
}

# start include (verbatim mode)
/<\?BEGININC .* \?>$/ {
    print
    keyword = "<\?BEGININC "
    extract = match($0, /<\?BEGININC .* \?>$/)
    includefile = substr($0, RSTART + length(keyword),
                        RLENGTH - 3 - length(keyword))
    output = filecontents(includefile)
    printf("%s", output)
}

# start include (escape-for-XML mode)
```

14. Informative References

- [BCP97] Klensin, J. and S. Hartman, "[Handling Normative References to Standards-Track Documents](#)", [BCP 97](#), RFC 4897, June 2007, <<https://www.rfc-editor.org/info/bcp>>.
- [DC-HTML] Johnston, P. and A. Powell, "[Expressing Dublin Core metadata using HTML/XHTML meta and link elements](#)", Dublin Core Metadata Initiative, August 2008, <<http://dublincore.org/documents/2008/08/04/dc-html/>>.
- [HTML5] Hickson, I., Berjon, R., Faulkner, S., Leithead, T., Doyle Navara, E., O'Connor, E., and S. Pfeiffer, "[HTML5](#)", W3C Recommendation REC-html5-20141028, October 2014, <<http://www.w3.org/TR/2014/REC-html5-20141028/>>. Latest version available at <<http://www.w3.org/TR/html5/>>.
- [RFC2026] Bradner, S., "[The Internet Standards Process -- Revision 3](#)", [BCP 9](#), RFC 2026, [DOI 10.17487/RFC2026](#), October 1996, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC2045] Freed, N. and N. Borenstein, "[Multipurpose Internet Mail Extensions \(MIME\) Part One: Format of Internet Message Bodies](#)", RFC 2045, [DOI 10.17487/RFC2045](#), November 1996, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC2119] Bradner, S., "[Key words for use in RFCs to Indicate Requirement Levels](#)", [BCP 14](#), RFC 2119, [DOI 10.17487/RFC2119](#), March 1997, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Nielsen, H., Masinter, L., Leach, P., and T. Berners-Lee, "[Hypertext Transfer Protocol -- HTTP/1.1](#)", RFC 2616, [DOI 10.17487/RFC2616](#), June 1999, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC2629] Rose, M., "[Writing I-Ds and RFCs using XML](#)", RFC 2629, [DOI 10.17487/RFC2629](#), June 1999, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC2648] Moats, R., "[A URN Namespace for IETF Documents](#)", RFC 2648, [DOI 10.17487/RFC2648](#), August 1999, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC2731] Kunze, J., "[Encoding Dublin Core Metadata in HTML](#)", RFC 2731, [DOI 10.17487/RFC2731](#), December 1999, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC5234] Crocker, D., Ed. and P. Overell, "[Augmented BNF for Syntax Specifications: ABNF](#)", [STD 68](#), RFC 5234, [DOI 10.17487/RFC5234](#), January 2008, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "[Hypertext Transfer Protocol \(HTTP/1.1\): Message Syntax and Routing](#)", RFC 7230, [DOI 10.17487/RFC7230](#), June 2014, <<https://www.rfc-editor.org/info/rfc>>.
- [RFC7749] Reschke, J., "[The 'xml2rfc' Version 2 Vocabulary](#)", RFC 7749, [DOI 10.17487/RFC7749](#), February 2016.
- [RFC7991bis] Levine, J., Ed., "[The 'xml2rfc' version 3 Vocabulary](#)", Internet-Draft draft-iab-rfc7991bis-03 ([work in progress](#)), March 2020.
- [RFC7991bis-01] Hoffman, P., "[The 'xml2rfc' version 3 Vocabulary](#)", Internet-Draft draft-iab-rfc7991bis-01 ([work in progress](#)), October 2018.
- [RNC] Clark, J., "[RELAX NG Compact Syntax](#)", OASIS, Nov 2002, <<http://www.oasis-open.org/committees/relax-ng/compact-20021121.html>>.
- [V3IMPNOTES] Levkowitz, H., "[Implementation notes for RFC7991, 'The 'xml2rfc' Version 3 Vocabulary'](#)", Internet-Draft draft-levkowitz-xml2rfc-v3-implementation-notes-12 ([work in progress](#)), December 2020.
- [XHTML2] Birbeck, M., Gylling, M., McCarron, S., Pemberton, S., Axelsson, J., Dubinko, M., Epperson, B., Ishikawa, M., and A. Navarro, "[XHTML\(tm\) 2.0](#)", W3C Group Note

NOTE-xhtml2-20101216, December 2010, <<http://www.w3.org/TR/2010/NOTE-xhtml2-20101216>>.

Latest version available at <<http://www.w3.org/TR/xhtml2>>.

[XML]

Bray, T., Paoli, J., Sperberg-McQueen, M., Maler, E., and F. Yergeau, "[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>>.

[XSL-FO]

Berglund, A., "[Extensible Stylesheet Language \(XSL\) Version 1.1](#)", W3C Recommendation REC-xsl11-20061205, December 2006, <<http://www.w3.org/TR/2006/REC-xsl11-20061205/>>.

Latest version available at <<http://www.w3.org/TR/xsl11/>>.

Appendix A. RELAX NG Compact Schema

The RelaxNG schema ([RNC]) below can be used to validate input documents (for instance, with [Jing](#)¹⁴).

Note that this is work in progress, and doesn't yet cover all extensions completely.

¹⁴ <http://www.thaiopensource.com/relaxng/jing.html>

```

# WORK IN PROGRESS! PLEASE REPORT PROBLEMS TO THE AUTHOR.

# Define our extension namespace
namespace x = "http://purl.org/net/xml2rfc/ext"

# Define GRDDL namespace
namespace grddl = "http://www.w3.org/2003/g/data-view#"

# Define RDF namespace
namespace rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"

# Define SVG namespace
namespace svg = "http://www.w3.org/2000/svg"

# Define XInclude namespace
namespace xi = "http://www.w3.org/2001/XInclude"

# Include rfc2629bis RNC grammar
include "rfc2629.rnc" {

  # Redefine <address> to allow multiple email addresses
  address =
    element address {
      attlist.address,
      postal?,
      phone?,
      facsimile?,
      email*,
      uri?
    }

  # Redefine <annotation> to allow more markup
  annotation =
    element annotation {
      attlist.annotation,
      (TEXT
        | xref
        | eref
        | iref
        | cref
        | spanx
        | v3_tt
      )*
    }

  # Redefine <artwork> to allow markup
  artwork =
    element artwork {
      attlist.artwork,
      attribute pn { text }?, # (see [RFC7991bis], Appendix B.2)
      ( v3_svg |
        (TEXT
          | eref
          | iref
          | spanx
          | xref
          | v3_em
          | v3_strong
          | x_abnf-char-sequence
          | x_bb

```

Appendix B. Implementation Notes

B.1. Recognized type attributes for <artwork> element

Specific values in the <artwork> element's "type" attribute are recognized and cause a different visual style to be used:

Type	Comment
abnf	ABNF as per [RFC5234]
abnf2045	ABNF as per [RFC2045]
abnf2616	ABNF as per [RFC2616] , Section 2.1
abnf7230	ABNF as per [RFC7230] , Section 1.2
application/relax-ng-compact-syntax	Relax NG Compact Syntax as per [RNC]
application/xml-dtd	XML DTD
code	monospaced text (with outline)
drawing	drawing (with outline)
example	monospaced text (with outline)
image/*	images (to be used with "src" attribute)
inline	monospaced text (no outline)
message/http; msgtype="request"	HTTP message, as per [RFC7230] , Section 8.3.1
message/http; msgtype="response"	HTTP message, as per [RFC7230] , Section 8.3.1
pdu	pdu (with outline)
svg and image/svg+xml	SVG

Appendix C. Examples

C.1. Using the 'Internal Subset'

The prolog of the XML document can both be used to refer to an external DTD, and also to define internal entities ([Section 2.8](#) of [XML]):

```
<?xml version="1.0"?>
<?xml-stylesheet type='text/xsl' href='rfc2629.xslt' ?>

<!DOCTYPE rfc SYSTEM "rfc2629.dtd" [

  <!-- use "&MAY;" for a BCP 14 "MAY", see Section 11.5 -->
  <!ENTITY MAY
    "<bcp14 xmlns='http://purl.org/net/xml2rfc/ext'>MAY</bcp14>">

  <!-- re-declare "&nbsp;" as code point 160 (non-breaking space) -->
  <!-- you may need this for UAs that do not read external DTDs -->
  <!ENTITY nbsp
    "&#160;">

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

  <!-- allow a custom reference using "&mydraft;" -->
  <!-- the data will be fetched from the same location as the
        source file -->
  <!ENTITY mydraft SYSTEM "reference.mydraft.xml">
]>
```

Note: including entities from a remote site will not work in browsers due to the Same-Origin policy.

C.2. Customization

The XSLT code can be customized by creating a custom XSLT file that uses `<xsl:import>` to include the original code, and just overrides particular rules.

For instance, the code below overrides several attributes in `rfc2629toFO.xslt`, changing the color, spacing and font family for headers.

```
<xsl:transform xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
               version="1.0">

  <xsl:import href="rfc2629toFO.xslt"/>

  <xsl:attribute-set name="h1">
    <xsl:attribute name="color">darkblue</xsl:attribute>
    <xsl:attribute name="font-family">sans-serif</xsl:attribute>
    <xsl:attribute name="space-before">24pt</xsl:attribute>
  </xsl:attribute-set>

  <xsl:attribute-set name="h2">
    <xsl:attribute name="color">darkblue</xsl:attribute>
    <xsl:attribute name="font-family">sans-serif</xsl:attribute>
    <xsl:attribute name="space-before">18pt</xsl:attribute>
    <xsl:attribute name="space-after">3pt</xsl:attribute>
  </xsl:attribute-set>

  <xsl:attribute-set name="h3">
    <xsl:attribute name="color">darkblue</xsl:attribute>
    <xsl:attribute name="font-family">sans-serif</xsl:attribute>
    <xsl:attribute name="space-before">16pt</xsl:attribute>
    <xsl:attribute name="space-after">2pt</xsl:attribute>
  </xsl:attribute-set>

</xsl:transform>
```

Note: the name for the attribute sets may change in the future as more working is done with respect to customizability. In any case, overriding the settings in a separate file will be easier to maintain. Please contact the author if you find yourself trying to override style definitions that currently do not use attribute sets.

Note: the CSS style information used in `rfc2629.xslt` can be overridden in a similar (but less granular) way: just overwrite the template called "insertCss". As for XSL-FO, the class names may change in future.

Appendix D. Producing the IETF 'Boilerplate'

Various attributes of the `<rfc>` element plus some child elements of `<front>` affect the automatically generated parts of the front page, such as the tabular information at the beginning, the "Status Of This Memo", and the "Copyright Notice".

When submitting an Internet Draft, this "boilerplate" is checked by "Idnits" (<http://tools.ietf.org/tools/idnits/>) for compliance with the current Intellectual Property rules, and thus it is important to set the correct values.

Furthermore, the RFC Production Center uses RFC7749-based tools to generate the final RFC text, so the more accurate the supplied information is, the less additional work is left, and the risk for errors in producing the final (and immutable!) document is reduced.

Note: this only applies to the case when IETF documents are produced. The "private" processing instruction `allows` to switch off most of the autogeneration logic.

The individual attributes on the `<rfc>` element are discussed in detail in [Appendix A](#) of [RFC7749].

Appendix E. License

Copyright (c) 2006-2019, Julian Reschke (julian.reschke@greenbytes.de)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Julian Reschke nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Index

A

- abnf-char-sequence Extension Element [29](#), [50](#)
- abort-on PI pseudo-attribute [10](#)
- allow-markup-in-artwork PI pseudo-attribute [11](#)
- alternate HTML LINK element [22](#)
- anchor-alias Extension Element [29](#), [50](#)
- Anchors
 - rfc.abstract [18](#)
 - rfc.authors [18](#)
 - rfc.copyright [18](#)
 - rfc.copyrightnotice [18](#)
 - rfc.figure.n [18](#), [18](#)
 - rfc.figure.u.n [18](#), [18](#)
 - rfc.index [18](#)
 - rfc.ipr [18](#)
 - rfc.iref.n [18](#)
 - rfc.note.n [18](#)
 - rfc.references [18](#), [18](#)
 - rfc.section.n [18](#)
 - rfc.section.n.p.m [18](#)
 - rfc.status [18](#)
 - rfc.toc [18](#)
 - rfc.xref.name.n [18](#)
- AntennaHouse XSL Formatter [27](#)
- Apache FOP [27](#)
- appendix HTML LINK element [22](#)
- artset V3 Extension Element [36](#), [50](#)
- aside V3 Extension Element [36](#), [50](#)
- assign-section-number Extension Element [50](#)
- author HTML LINK element [22](#)
- authors-section PI pseudo-attribute [11](#)
- authorship PI pseudo-attribute [9](#)

B

- background PI pseudo-attribute [9](#)
- bb Extension Element [29](#), [50](#)
- bc Extension Element [29](#), [50](#)
- bcp14 Extension Element [29](#), [50](#)
- bcp14 V3 Extension Element [36](#), [50](#)
- BCP97* [40](#), [47](#)
- blockquote Extension Element [30](#), [50](#)
- blockquote V3 Extension Element [36](#), [50](#)
- boilerplate V3 Extension Element [36](#), [50](#)
- br V3 Extension Element [36](#)
- bt Extension Element [30](#), [50](#)

C

- c Extension Element [30](#), [50](#)
- chapter HTML LINK element [22](#)
- CHM format [26](#)
- cityarea V3 Extension Element [36](#), [50](#)
- comments PI pseudo-attribute [9](#)
- compact PI pseudo-attribute [9](#)
- contact V3 Extension Element [36](#), [50](#)
- contents HTML LINK element [22](#)
- contributor Extension Element [30](#), [50](#)
- copyright HTML LINK element [22](#)

- Creator DCMI property [23](#)
- css-contents PI pseudo-attribute [11](#)
- css-resource PI pseudo-attribute [11](#)

D

- dark-mode PI pseudo-attribute [11](#)
- Date.Issued DCMI property [23](#)
- DC-HTML* [23](#), [47](#)
- DCMI properties
 - Creator [23](#)
 - Date.Issued [23](#)
 - Description.Abstract [23](#)
 - Identifier [23](#)
 - isPartOf [23](#)
 - Relation.Replaces [23](#)
- dd V3 Extension Element [36](#), [50](#)
- defines Extension Element [50](#)
- Description.Abstract DCMI property [23](#)
- dfn Extension Element [31](#), [50](#)
- diff-uri PI pseudo-attribute [12](#)
- displayreference V3 Extension Element [36](#), [50](#)
- dl V3 Extension Element [36](#), [50](#)
- docmapping PI pseudo-attribute [9](#)
- doi-uri PI pseudo-attribute [12](#)
- dt V3 Extension Element [36](#), [50](#)
- duplex PI pseudo-attribute [12](#)

E

- editing PI pseudo-attribute [9](#)
- em V3 Extension Element [37](#), [50](#), [50](#)
- errata PI pseudo-attribute [12](#), [24](#)
- extaddr V3 Extension Element [37](#), [50](#), [50](#)
- Extension Elements
 - abnf-char-sequence [29](#), [50](#)
 - anchor-alias [29](#), [50](#)
 - assign-section-number [50](#)
 - bb [29](#), [50](#)
 - bc [29](#), [50](#)
 - bcp14 [29](#), [50](#)
 - blockquote [30](#), [50](#)
 - bt [30](#), [50](#)
 - c [30](#), [50](#)
 - contributor [30](#), [50](#)
 - defines [50](#)
 - dfn [31](#), [50](#)
 - feedback [31](#), [50](#)
 - h [31](#), [50](#)
 - has [50](#)
 - highlight [31](#), [50](#)
 - include-author [50](#)
 - length-of [31](#), [50](#)
 - link [31](#), [50](#)
 - lt [32](#), [50](#)
 - note [32](#), [50](#)
 - parse-xml [32](#), [50](#)
 - prose [32](#), [50](#)
 - q [32](#), [50](#)
 - rdf:Description [50](#)
 - ref [32](#), [50](#)
 - source [33](#), [50](#)

span 50
 sup 33, 50
 u-map 33, 50
 x 50

F

feedback Extension Element 31, 50
 Firefox
 1.* / 2.* 19
 3.* 19
 footer PI pseudo-attribute 9

G

generator HTML META element 23
 Google Chrome 19
 Grammar 49

H

h Extension Element 31, 50
 has Extension Element 50
 header PI pseudo-attribute 9
 highlight Extension Element 31, 50
 HTML compliance 21
 HTML LINK elements
 alternate 22
 appendix 22
 author 22
 chapter 22
 contents 22
 copyright 22
 index 22
 HTML META elements
 generator 23
 keywords 23
 html-pretty-print PI pseudo-attribute 12
HTML5 21, 21, 30, 31, 32, 33, 47
 Section 4.4.4 30
 Section 4.5.16 33
 Section 4.5.7 32
 Section 4.5.8 31
 Section 8.1.1 21

I

Identifier DCMI property 23
 include PI pseudo-attribute 9, 45
 include-author Extension Element 50
 include-generator PI pseudo-attribute 12
 include-index PI pseudo-attribute 13
 include-references-in-index PI pseudo-attribute 13
 index HTML LINK element 22
 inline PI pseudo-attribute 9
 insert-metadata PI pseudo-attribute 13, 23
 Internet Explorer 6 19
 Internet Explorer 7 19
 Internet Explorer 8 19
 Internet Explorer 9 19
 internet-draft-reference-base-uri PI pseudo-attribute 13
 internet-draft-uri PI pseudo-attribute 13
 iprnotified PI pseudo-attribute 9
 isbn-uri PI pseudo-attribute 14

isPartOf DCMI property 23

J

justification PI pseudo-attribute 14

K

keywords HTML META element 23

L

length-of Extension Element 31, 50
 li V3 Extension Element 37, 50
 link Extension Element 31, 50
 link V3 Extension Element 37, 50
 linkmailto PI pseudo-attribute 9
 log-level PI pseudo-attribute 14
 lt Extension Element 32, 50

M

maxwidth PI pseudo-attribute 14
 Microsoft Help 26
 Mozilla 19, 19
 MSXML3 19
 MSXML4 19
 multiple-initials PI pseudo-attribute 9

N

name V3 Extension Element 37, 50
 needLines PI pseudo-attribute 10
 NoScript 19
 note Extension Element 32, 50

O

ol V3 Extension Element 37, 50
 Opera 19

P

Parameters
 xml2rfc-authorship 9
 xml2rfc-background 9
 xml2rfc-comments 9
 xml2rfc-compact 9
 xml2rfc-editing 9
 xml2rfc-ext-abort-on 10
 xml2rfc-ext-allow-markup-in-artwork 11
 xml2rfc-ext-authors-section 11
 xml2rfc-ext-css-contents 11
 xml2rfc-ext-css-resource 11
 xml2rfc-ext-dark-mode 11
 xml2rfc-ext-diff-uri 12
 xml2rfc-ext-doi-uri 12
 xml2rfc-ext-duplex 12
 xml2rfc-ext-errata 12, 24
 xml2rfc-ext-html-pretty-print 12
 xml2rfc-ext-include-generator 12
 xml2rfc-ext-include-index 13
 xml2rfc-ext-include-references-in-index 13
 xml2rfc-ext-insert-metadata 13, 23
 xml2rfc-ext-internet-draft-uri 13
 xml2rfc-ext-isbn-uri 14
 xml2rfc-ext-justification 14
 xml2rfc-ext-log-level 14

- xml2rfc-ext-maxwidth [14](#)
 - xml2rfc-ext-parse-xml-in-artwork [14](#)
 - xml2rfc-ext-rfc-errata-uri [14](#)
 - xml2rfc-ext-rfc-erratum-uri [15](#)
 - xml2rfc-ext-rfc-reference-base-uri [13](#), [15](#)
 - xml2rfc-ext-rfc-uri [15](#)
 - xml2rfc-ext-sec-no-trailing-dots [15](#)
 - xml2rfc-ext-std-uri [15](#)
 - xml2rfc-ext-support-highwire-press-tags [15](#)
 - xml2rfc-ext-support-open-graph-tags [16](#)
 - xml2rfc-ext-support-rfc2731 [16](#)
 - xml2rfc-ext-ucd-file [16](#)
 - xml2rfc-ext-xml2rfc-backend [16](#)
 - xml2rfc-ext-xref-with-text-generate [16](#)
 - xml2rfc-footer [9](#)
 - xml2rfc-header [9](#)
 - xml2rfc-inline [9](#)
 - xml2rfc-iprnotified [9](#)
 - xml2rfc-linkmailto [9](#)
 - xml2rfc-multiple-initials [9](#)
 - xml2rfc-private [9](#)
 - xml2rfc-refparent [9](#)
 - xml2rfc-rfcstyle [9](#)
 - xml2rfc-sortrefs [9](#)
 - xml2rfc-symrefs [9](#)
 - xml2rfc-toc [9](#)
 - xml2rfc-tocdepth [9](#)
 - xml2rfc-topblock [9](#)
 - parse-xml Extension Element [32](#), [50](#)
 - parse-xml-in-artwork PI pseudo-attribute [14](#)
 - pobox V3 Extension Element [37](#)
 - postalLine V3 Extension Element [37](#), [50](#)
 - private PI pseudo-attribute [9](#), [54](#)
 - Processing Instruction pseudo attributes
 - abort-on [10](#)
 - allow-markup-in-artwork [11](#)
 - authors-section [11](#)
 - authorship [9](#)
 - background [9](#)
 - comments [9](#)
 - compact [9](#)
 - css-contents [11](#)
 - css-resource [11](#)
 - dark-mode [11](#)
 - diff-uri [12](#)
 - docmapping [9](#)
 - doi-uri [12](#)
 - duplex [12](#)
 - editing [9](#)
 - errata [12](#), [24](#)
 - footer [9](#)
 - header [9](#)
 - html-pretty-print [12](#)
 - include [9](#), [45](#)
 - include-generator [12](#)
 - include-index [13](#)
 - include-references-in-index [13](#)
 - inline [9](#)
 - insert-metadata [13](#), [23](#)
 - internet-draft-reference-base-uri [13](#)
 - internet-draft-uri [13](#)
 - iprnotified [9](#)
 - isbn-uri [14](#)
 - justification [14](#)
 - linkmailto [9](#)
 - log-level [14](#)
 - maxwidth [14](#)
 - multiple-initials [9](#)
 - needLines [10](#)
 - parse-xml-in-artwork [14](#)
 - private [9](#), [54](#)
 - refparent [9](#)
 - rfc-errata-uri [14](#)
 - rfc-erratum-uri [15](#)
 - rfc-reference-base-uri [15](#)
 - rfc-uri [15](#)
 - rfcstyle [9](#)
 - sec-no-trailing-dots [15](#)
 - slides [10](#)
 - sortrefs [9](#)
 - std-uri [15](#)
 - strict [10](#)
 - subcompact [10](#)
 - support-highwire-press-tags [15](#)
 - support-open-graph-tags [16](#)
 - support-rfc2731 [16](#)
 - symrefs [9](#)
 - toc [9](#)
 - tocdepth [9](#)
 - tocindent [10](#)
 - tocompact [10](#)
 - topblock [9](#)
 - ucd-file [16](#)
 - xml2rfc-backend [16](#)
 - xref-with-text-generate [16](#)
 - prose Extension Element [32](#), [50](#)
- ## Q
- q Extension Element [32](#), [50](#)
- ## R
- rdf:Description Extension Element [50](#)
 - ref Extension Element [32](#), [50](#)
 - refcontent V3 Extension Element [37](#), [50](#)
 - reference V3 Extension Element [37](#)
 - referencegroup V3 Extension Element [37](#), [50](#)
 - refparent PI pseudo-attribute [9](#)
 - Relation.Replaces DCMI property [23](#)
 - RELAX NG Compact Schema [49](#)
 - relref V3 Extension Element [37](#), [50](#)
 - rfc-errata-uri PI pseudo-attribute [14](#)
 - rfc-erratum-uri PI pseudo-attribute [15](#)
 - rfc-reference-base-uri PI pseudo-attribute [15](#)
 - rfc-uri PI pseudo-attribute [15](#)
 - rfc.abstract anchor [18](#)
 - rfc.authors anchor [18](#)
 - rfc.copyright anchor [18](#)
 - rfc.copyrightnotice anchor [18](#)
 - rfc.figure.n anchor [18](#), [18](#)
 - rfc.figure.u.n anchor [18](#), [18](#)

[rfc.index](#) anchor 18
[rfc.ipr](#) anchor 18
[rfc.iref.n](#) anchor 18
[rfc.note.n](#) anchor 18
[rfc.references](#) anchor 18
[rfc.references.n](#) anchor 18
[rfc.section.n](#) anchor 18
[rfc.section.n.p.m](#) anchor 18
[rfc.status](#) anchor 18
[rfc.toc](#) anchor 18
[rfc.xref.name.n](#) anchor 18
[RFC2026](#) 34, 47
 Section 4.1 34
[RFC2045](#) 47, 51
[RFC2119](#) 29, 35, 47
 Section 5 35
[RFC2616](#) 47, 51
 Section 2.1 51
[RFC2629](#) 7, 47
[RFC2648](#) 23, 47
[RFC2731](#) 23, 47
[RFC5234](#) 47, 51
[RFC7230](#) 47, 51, 51, 51
 Section 1.2 51
 Section 8.3.1 51, 51
[RFC7749](#) 7, 8, 47, 54
 Appendix A 54
[RFC7991bis](#) 8, 36, 36, 36, 36, 36, 36, 36, 36, 37, 37, 37,
 37, 37, 37, 37, 37, 37, 37, 37, 38, 38, 38, 38, 38, 38, 38, 38,
 38, 38, 38, 39, 39, 39, 39, 39, 39, 39, 39, 44, 47, 50, 50, 50, 50,
 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
 50, 50, 50, 50, 50
 Section 2 50
 Section 2.1.1 50
 Section 2.7 36
 Section 2.10 36
 Section 2.11 36, 36
 Section 2.17.2 50
 Section 2.19 36
 Section 2.20 36
 Section 2.21 36
 Section 2.21.1 50
 Section 2.22 37
 Section 2.23 37
 Section 2.31 37
 Section 2.32 37
 Section 2.34 37
 Section 2.36 37
 Section 2.41 37
 Section 2.42 37
 Section 2.43 37
 Section 2.44 37
 Section 2.47 37
 Section 2.48.4 50
 Section 2.48.11 50
 Section 2.48.13 50, 50
 Section 2.48.14 50
 Section 2.48.15 50
 Section 2.49 38

 Section 2.49.2 38
 Section 2.49.3 38
 Section 2.52 38
 Section 2.54 38
 Section 2.55 38
 Section 2.56 38
 Section 2.58 38
 Section 2.59 38
 Section 2.60 38
 Section 2.61 39
 Section 2.62 39
 Section 2.63 39
 Section 2.65 39
 Section 2.66 39
 Section 2.67 39
 Section 4 38
 Appendix B.1 39
 Appendix B.2 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50, 50,
 50
 Appendix B.3 50, 50, 50, 50
[RFC7991bis-01](#) 37, 47
 Section 2.39.2 37
[rfcdestyle](#) PI pseudo-attribute 9
[RNC](#) 47, 49, 51

S

[Safari](#) 19, 19
 3.* 19
[Saxon](#) 19
[Schema](#) 49
[sec-no-trailing-dots](#) PI pseudo-attribute 15
[section](#) V3 Extension Element 38
[slides](#) PI pseudo-attribute 10
[sortingcode](#) V3 Extension Element 38, 50
[sortrefs](#) PI pseudo-attribute 9
[source](#) Extension Element 33, 50
[sourcecode](#) V3 Extension Element 38, 50
[span](#) Extension Element 50
[std-uri](#) PI pseudo-attribute 15
[strict](#) PI pseudo-attribute 10
[strong](#) V3 Extension Element 38, 50
[sub](#) V3 Extension Element 38, 50
[subcompact](#) PI pseudo-attribute 10
[sup](#) Extension Element 33, 50
[sup](#) V3 Extension Element 38, 50
[support-highwire-press-tags](#) PI pseudo-attribute 15
[support-open-graph-tags](#) PI pseudo-attribute 16
[support-rfc2731](#) PI pseudo-attribute 16
[svg](#) V3 Extension Element 38, 50
[symrefs](#) PI pseudo-attribute 9

T

[table](#) V3 Extension Element 38, 50
[tbody](#) V3 Extension Element 38, 50
[td](#) V3 Extension Element 38, 50
[tfoot](#) V3 Extension Element 38, 50
[th](#) V3 Extension Element 39, 50
[thead](#) V3 Extension Element 39, 50
[toc](#) PI pseudo-attribute 9

tocdepth PI pseudo-attribute 9
 tocindent PI pseudo-attribute 10
 tocompact PI pseudo-attribute 10
 topblock PI pseudo-attribute 9
 tr V3 Extension Element 39
 tre V3 Extension Element 50
 tt V3 Extension Element 39, 50

U

u V3 Extension Element 39, 50
 u-map Extension Element 33, 50
 ucd-file PI pseudo-attribute 16
 ul V3 Extension Element 39, 50

V

V3 Extension Elements

artset 36, 50
 aside 36, 50
 bcp14 36, 50
 blockquote 36, 50
 boilerplate 36, 50
 br 36, 50
 cityarea 36, 50
 contact 36, 50
 dd 36, 50
 displayreference 36, 50
 dl 36, 50
 dt 36, 50
 em 37, 50
 extaddr 37, 50
 li 37, 50
 link 37, 50
 name 37, 50
 ol 37, 50
 pobox 37, 50
 postalLine 37, 50
 refcontent 37, 50
 reference 37
 referencegroup 37, 50
 relref 37, 50
 section 38
 sortingcode 38, 50
 sourcecode 38, 50
 strong 38, 50
 sub 38, 50
 sup 38, 50
 svg 38, 50
 table 38, 50
 tbody 38, 50
 td 38, 50
 tfoot 38, 50
 th 39, 50
 thead 39, 50
 tr 39, 50
 tt 39, 50
 u 39, 50
 ul 39, 50
 xi:include 39
 xref 39
 V3IMPNOTES 34, 36, 36, 37, 37, 38, 39, 47, 50, 50

Appendix A.1 39
 Section 3.1.1 36
 Section 3.1.16 50
 Section 3.1.22 50
 Section 4.1.3.2 34
 Section 4.2.7 36, 37, 38

X

x Extension Element 50
 Xalan 19
 XHTML2 31, 47
 xi:include Extension Element 50
 xi:include V3 Extension Element 39
 XInclude Extension Elements
 include 50
 XML 48, 52
 Section 2.8 52
 xml-styleSheet PI 21
 xml2rfc-authorship parameter 9
 xml2rfc-backend PI pseudo-attribute 16
 xml2rfc-background parameter 9
 xml2rfc-comments parameter 9
 xml2rfc-editing parameter 9, 9
 xml2rfc-ext-abort-on parameter 10
 xml2rfc-ext-allow-markup-in-artwork parameter 11
 xml2rfc-ext-authors-section parameter 11
 xml2rfc-ext-css-contents parameter 11
 xml2rfc-ext-css-resource parameter 11
 xml2rfc-ext-dark-mode parameter 11
 xml2rfc-ext-diff-uri parameter 12
 xml2rfc-ext-doi-uri parameter 12
 xml2rfc-ext-duplex parameter 12
 xml2rfc-ext-errata 24
 xml2rfc-ext-errata parameter 12
 xml2rfc-ext-html-pretty-print parameter 12
 xml2rfc-ext-include-generator parameter 12
 xml2rfc-ext-include-index parameter 13
 xml2rfc-ext-include-references-in-index parameter 13
 xml2rfc-ext-insert-metadata parameter 13, 23
 xml2rfc-ext-internet-draft-uri parameter 13
 xml2rfc-ext-isbn-uri parameter 14
 xml2rfc-ext-justification parameter 14
 xml2rfc-ext-log-level parameter 14
 xml2rfc-ext-maxwidth parameter 14
 xml2rfc-ext-parse-xml-in-artwork parameter 14
 xml2rfc-ext-rfc-errata-uri parameter 14
 xml2rfc-ext-rfc-erratum-uri parameter 15
 xml2rfc-ext-rfc-reference-base-uri 13, 15
 xml2rfc-ext-rfc-uri parameter 15
 xml2rfc-ext-sec-no-trailing-dots parameter 15
 xml2rfc-ext-std-uri parameter 15
 xml2rfc-ext-support-highwire-press-tags parameter 15
 xml2rfc-ext-support-open-graph-tags parameter 16
 xml2rfc-ext-support-rfc2731 parameter 16
 xml2rfc-ext-ucd-file parameter 16
 xml2rfc-ext-xml2rfc-backend parameter 16
 xml2rfc-ext-xref-with-text-generate parameter 16
 xml2rfc-footer parameter 9
 xml2rfc-header parameter 9
 xml2rfc-inline parameter 9

- xml2rfc-iprnotified parameter [9](#)
- xml2rfc-linkmailto parameter [9](#)
- xml2rfc-multiple-initials parameter [9](#)
- xml2rfc-private parameter [9](#)
- xml2rfc-refparent parameter [9](#)
- xml2rfc-rfcedstyle parameter [9](#)
- xml2rfc-sortrefs parameter [9](#)
- xml2rfc-symrefs parameter [9](#)
- xml2rfc-toc parameter [9](#)
- xml2rfc-tocdepth parameter [9](#)
- xml2rfc-topblock parameter [9](#)
- xref V3 Extension Element [39](#)
- xref-with-text-generate PI pseudo-attribute [16](#)
- XSL-FO* [27](#), [27](#), [48](#)
- xsltproc [9](#)
 - passing parameters [9](#)

Author's Address

Julian F. Reschke

greenbytes GmbH

Hafenweg 16

Muenster, NW 48155

Germany

E-Mail: julian.reschke@greenbytes.de

URI: <https://greenbytes.de/tech/webdav/>