Transforming RFC2629-formatted XML through XSLT
Table of Contents

1 Introduction .............................................................................................................................................................. 5

2 Supported RFC2629 elements .................................................................................................................................. 6
  2.1 Extension elements .................................................................................................................................................. 6

3 Processing Instructions ............................................................................................................................................... 7
  3.1 Supported xml2rfc-compatible PIs ......................................................................................................................... 7
  3.2 Unsupported xml2rfc-compatible PIs ......................................................................................................................... 8
  3.3 Extension PIs .......................................................................................................................................................... 8

4 Anchors ..................................................................................................................................................................... 10

5 Supported XSLT engines .......................................................................................................................................... 11
  5.1 Standalone Engines .................................................................................................................................................. 11
  5.2 In-Browser Engines ................................................................................................................................................. 11

6 Transforming to HTML .............................................................................................................................................. 13
  6.1 HTML compliance ..................................................................................................................................................... 13
  6.2 Standard HTML LINK elements .............................................................................................................................. 13
  6.3 Standard HTML metadata ....................................................................................................................................... 13
  6.4 Dublin Core (RFC2731) metadata .......................................................................................................................... 14

7 Transforming to XHTML ........................................................................................................................................ 15

8 Transforming to CHM (Microsoft Compiled Help) .................................................................................................. 16

9 Transforming to PDF .................................................................................................................................................. 17
  9.1 Via XSL-FO ............................................................................................................................................................. 17
    9.1.1 Example: producing output for Apache FOP .................................................................................................. 17
  9.2 Via X(HTML) ......................................................................................................................................................... 17

10 Transforming to ePub ............................................................................................................................................ 18

11 Generic Extensions ..................................................................................................................................................... 19
  11.1 <abnf-char-sequence> element .............................................................................................................................. 19
  11.2 <anchor-alias> element .......................................................................................................................................... 19
  11.3 <bcp14> element ..................................................................................................................................................... 19
  11.4 <bb> element ........................................................................................................................................................ 19
  11.5 <bc> element ......................................................................................................................................................... 19
  11.6 <blockquote> element .............................................................................................................................................. 20
  11.7 <boilerplate> element ............................................................................................................................................ 20
  11.8 <bt> element ........................................................................................................................................................ 20
  11.9 <dfn> element ....................................................................................................................................................... 20
  11.10 <feedback> element ......................................................................................................................................... 20

Documentation  RFC2629 through XSLT  September 2015
11.11 <h> element........................................................................................................................................................... 20
11.12 <highlight> element................................................................................................................................................. 20
11.13 <length-of> element................................................................................................................................................. 20
11.14 <link> element...................................................................................................................................................... 21
11.15 <lt> element.......................................................................................................................................................... 21
11.16 <note> element...................................................................................................................................................... 21
11.17 <parse-xml> element............................................................................................................................................. 21
11.18 <prose> element.................................................................................................................................................... 21
11.19 <q> element........................................................................................................................................................ 21
11.20 <ref> element....................................................................................................................................................... 22
11.21 <source> element................................................................................................................................................ 22
11.22 <sup> element....................................................................................................................................................... 22
11.23 Extensions to Xml2rfc <artwork> element........................................................................................................ 23
11.24 Extensions to Xml2rfc <ref> element.................................................................................................................. 23
11.25 Extensions to Xml2rfc <list> element.................................................................................................................. 23
11.26 Extensions to Xml2rfc <rfc> element.................................................................................................................... 23
11.27 Extensions to Xml2rfc <section> element.......................................................................................................... 23
11.28 Extensions to Xml2rfc <xref> element.................................................................................................................. 23

12 Experimental Support for XML2RFCv3 Vocabulary................................................................................................ 25
12.1 aside Element....................................................................................................................................................... 25
12.2 bcp14 Element....................................................................................................................................................... 25
12.3 blockquote Element.............................................................................................................................................. 25
12.4 displayreference Element.................................................................................................................................. 25
12.5 dd Element.......................................................................................................................................................... 25
12.6 dl Element.......................................................................................................................................................... 25
12.7 dt Element.......................................................................................................................................................... 25
12.8 em Element....................................................................................................................................................... 25
12.9 li Element.......................................................................................................................................................... 25
12.10 name Element................................................................................................................................................ 25
12.11 ol Element....................................................................................................................................................... 25
12.12 postalLine Element........................................................................................................................................... 25
12.13 refcontent Element........................................................................................................................................... 25
12.14 Extensions to reference Element...................................................................................................................... 26
12.14.1 quoteTitle attribute...................................................................................................................................... 26
12.15 Extensions to section Element............................................................................................................................ 26
12.15.1 numbered attribute...................................................................................................................................... 26
12.15.2 removeInRFC attribute.................................................................................................................................. 26
12.16 strong Element.................................................................................................................................................. 26
12.17 sub Element....................................................................................................................................................... 26
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.18</td>
<td>sup Element</td>
<td>26</td>
</tr>
<tr>
<td>12.19</td>
<td>tt Element</td>
<td>26</td>
</tr>
<tr>
<td>12.20</td>
<td>ul Element</td>
<td>26</td>
</tr>
<tr>
<td>12.21</td>
<td>Extensions to xref Element</td>
<td>26</td>
</tr>
<tr>
<td>12.21.1</td>
<td>relative attribute</td>
<td>26</td>
</tr>
<tr>
<td>12.21.2</td>
<td>section attribute</td>
<td>26</td>
</tr>
<tr>
<td>12.21.3</td>
<td>sectionFormat attribute</td>
<td>26</td>
</tr>
<tr>
<td>13</td>
<td>Utilities</td>
<td>28</td>
</tr>
<tr>
<td>13.1</td>
<td>Checking References</td>
<td>28</td>
</tr>
<tr>
<td>13.2</td>
<td>Generating Graphs from References</td>
<td>29</td>
</tr>
<tr>
<td>13.3</td>
<td>Producing reference entries for books</td>
<td>30</td>
</tr>
<tr>
<td>13.4</td>
<td>Down-converting to RFC2629bis DTD</td>
<td>31</td>
</tr>
<tr>
<td>13.5</td>
<td>Extracting artwork</td>
<td>31</td>
</tr>
<tr>
<td>13.6</td>
<td>GRRDL</td>
<td>32</td>
</tr>
<tr>
<td>13.7</td>
<td>HTML Live Refresh</td>
<td>32</td>
</tr>
<tr>
<td>14</td>
<td>Informative References</td>
<td>34</td>
</tr>
<tr>
<td>A</td>
<td>RELAX NG Compact Schema</td>
<td>36</td>
</tr>
<tr>
<td>B</td>
<td>Implementation Notes</td>
<td>38</td>
</tr>
<tr>
<td>B.1</td>
<td>Recognized type attributes for &lt;artwork&gt; element</td>
<td>38</td>
</tr>
<tr>
<td>C</td>
<td>Examples</td>
<td>39</td>
</tr>
<tr>
<td>C.1</td>
<td>Using the 'Internal Subset'</td>
<td>39</td>
</tr>
<tr>
<td>C.2</td>
<td>Customization</td>
<td>39</td>
</tr>
<tr>
<td>D</td>
<td>Producing the IETF 'Boilerplate'</td>
<td>41</td>
</tr>
<tr>
<td>D.1</td>
<td>The /rfc/@ipr Attribute</td>
<td>41</td>
</tr>
<tr>
<td>D.1.1</td>
<td>Current Values: &quot;trust200902&quot;</td>
<td>41</td>
</tr>
<tr>
<td>D.1.2</td>
<td>Historic Values</td>
<td>42</td>
</tr>
<tr>
<td>D.2</td>
<td>The /rfc/@category Attribute</td>
<td>43</td>
</tr>
<tr>
<td>D.3</td>
<td>The /rfc/@submissionType Attribute</td>
<td>43</td>
</tr>
<tr>
<td>D.4</td>
<td>The /rfc/@consensus Attribute</td>
<td>43</td>
</tr>
<tr>
<td>D.5</td>
<td>The /rfc/@number Attribute</td>
<td>43</td>
</tr>
<tr>
<td>D.6</td>
<td>The /rfc/@docName Attribute</td>
<td>43</td>
</tr>
<tr>
<td>D.7</td>
<td>The /rfc/@obsoletes Attribute</td>
<td>43</td>
</tr>
<tr>
<td>D.8</td>
<td>The /rfc/@updates Attribute</td>
<td>44</td>
</tr>
<tr>
<td>E</td>
<td>License</td>
<td>45</td>
</tr>
<tr>
<td>Index</td>
<td></td>
<td>46</td>
</tr>
<tr>
<td>Author's Address</td>
<td></td>
<td>50</td>
</tr>
</tbody>
</table>
1. Introduction

This document describes a set of XSLT transformations that can be used to transform "XML2RFC" XML ([XML2RFCV2], 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).
- various utilities (Section 13).

The full distribution is available at <http://greenbytes.de/tech/webdav/rfc2629xslt.zip>.
2. Supported RFC2629 elements

rfc2629.xslt supports both all RFC2629 grammar elements and the extensions implemented in xml2rfc 1.36.

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:

```xml
<?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

<table>
<thead>
<tr>
<th>PI target</th>
<th>PI pseudo-attribute</th>
<th>XSLT parameter name</th>
<th>default</th>
<th>comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>rfc</td>
<td>background</td>
<td>xml2rfc-background</td>
<td>(not set)</td>
<td>only applies to HTML output method when printing</td>
</tr>
<tr>
<td>rfc</td>
<td>compact</td>
<td>xml2rfc-compact</td>
<td>&quot;no&quot;</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>comments</td>
<td>xml2rfc-comments</td>
<td>(not set)</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>editing</td>
<td>xml2rfc-editing</td>
<td>&quot;no&quot;</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>footer</td>
<td>xml2rfc-footer</td>
<td>(not set)</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>header</td>
<td>xml2rfc-header</td>
<td>(not set)</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>inline</td>
<td>xml2rfc-inline</td>
<td>(not set)</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>iprnotified</td>
<td>xml2rfc-iprnotified</td>
<td>&quot;no&quot;</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>linkmailto</td>
<td>xml2rfc-linkmailto</td>
<td>&quot;yes&quot;</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>private</td>
<td>xml2rfc-private</td>
<td>(not set)</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>refparent</td>
<td>xml2rfc-private</td>
<td>&quot;References&quot;</td>
<td>Title for References sections when automatically inserted</td>
</tr>
<tr>
<td>rfc</td>
<td>rfredstyle</td>
<td>xml2rfc-rfredstyle</td>
<td>(not set)</td>
<td>(limited support)</td>
</tr>
<tr>
<td>rfc</td>
<td>sortrefs</td>
<td>xml2rfc-sortrefs</td>
<td>&quot;no&quot;</td>
<td>The default has changed from &quot;no&quot; to &quot;yes&quot; as of June 6, 2007 and xml2rfc 1.33pre4.</td>
</tr>
<tr>
<td>rfc</td>
<td>symrefs</td>
<td>xml2rfc-symrefs</td>
<td>&quot;yes&quot;</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>toc</td>
<td>xml2rfc-toc</td>
<td>&quot;no&quot;</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>tocdepth</td>
<td>xml2rfc-tocdepth</td>
<td>99</td>
<td></td>
</tr>
</tbody>
</table>
### 3.2 Unsupported xml2rfc-compatible PIs

<table>
<thead>
<tr>
<th>PI target</th>
<th>PI pseudo-attribute</th>
<th>XSLT parameter name</th>
<th>default</th>
<th>comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>rfc</td>
<td>include</td>
<td>xml2rfc-topblock</td>
<td>&quot;yes&quot;</td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>needLines</td>
<td>xml2rfc-topblock</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>slides</td>
<td>xml2rfc-topblock</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>strict</td>
<td>xml2rfc-topblock</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>subcompact</td>
<td>xml2rfc-topblock</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>tocindent</td>
<td>xml2rfc-topblock</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rfc</td>
<td>tocompact</td>
<td>xml2rfc-topblock</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Incompatible with XML/XSLT processing model, please use external entities instead (see Appendix C.1)

(defaults to "yes")

### 3.3 Extension PIs

<table>
<thead>
<tr>
<th>PI target</th>
<th>PI pseudo-attribute</th>
<th>XSLT parameter name</th>
<th>default</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rfc-ext</td>
<td>allow-markup-in-artwork</td>
<td>xml2rfc-allow-markup-in-artwork</td>
<td>&quot;no&quot;</td>
<td>Enables support for specific elements inside abstract elements (using this extension makes the document incompatible to the RFC2629bis DTD; see description of conversion XSLT in Section 13.4).</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>authors-section</td>
<td>xml2rfc-ext-authors-section</td>
<td>&quot;end&quot;</td>
<td>When &quot;before-appendices&quot;, place the authors section between references and appendices (this ordering was used a long time ago).</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>duplex</td>
<td>xml2rfc-ext-duplex</td>
<td>&quot;no&quot;</td>
<td>When set to &quot;yes&quot;, format the PDF output for doublesided printing.</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>include-index</td>
<td>xml2rfc-ext-index</td>
<td>&quot;yes&quot;</td>
<td>When set to &quot;no&quot;, no index will be generated.</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>include-references-in-index</td>
<td>xml2rfc-ext-index</td>
<td>&quot;no&quot;</td>
<td>When set to &quot;yes&quot;, index entries are generated for all references.</td>
</tr>
<tr>
<td>PI target</td>
<td>PI pseudo-attribute</td>
<td>XSLT parameter name</td>
<td>default</td>
<td>description</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------</td>
<td>--------------------------------------</td>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>insert-metadata</td>
<td>xml2rfc-ext-insert-metadata</td>
<td>&quot;yes&quot;</td>
<td>When set to &quot;yes&quot;, include JS code that fetches current RFC metadata and inserts it into the front page (standards track, obsoletion, updates, errata).</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>justification</td>
<td>xml2rfc-ext-justification</td>
<td>&quot;never&quot;</td>
<td>&quot;never&quot;: never emit justified text, &quot;always&quot;: always emit justified text, &quot;print&quot;: only emit justified text for print media.</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>parse-xml-in-artwork</td>
<td>xml2rfc-parse-xml-in-artwork</td>
<td>&quot;no&quot;</td>
<td>May be used to enable parsing of XML content in figures (MSXML only).</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>sec-no-trailing-dots</td>
<td>xml2rfc-ext-sec-no-trailing-dots</td>
<td></td>
<td>When set to &quot;yes&quot;, add trailing dots to section numbers. This seems to be the preferred format in the newest RFCs.</td>
</tr>
<tr>
<td>rfc-ext</td>
<td>support-rfc2731</td>
<td>xml2rfc-ext-support-rfc2731</td>
<td>&quot;yes&quot;</td>
<td>Decides whether the HTML transformation should generate META tags according Section 6.4.</td>
</tr>
</tbody>
</table>
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:

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

The transformation requires a non-standard extension function (see \texttt{exsl:node-set}) 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 \texttt{exsl:date-time} extension function.

5.1 Standalone Engines

The following XSLT engines are believed to work well:

- Windows: MSXML3 and MSXML4 (\url{http://msdn.microsoft.com/xml}); command line processor "msxsl" is available from Microsoft Download Center
- Java: Saxon (\url{http://saxon.sourceforge.net/})
- Java: Xalan (\url{http://xml.apache.org/xalan-j/})
- C/C++: xsltproc (libxslt) (\url{http://xmlsoft.org/XSLT/}), make sure that you have a current version

5.2 In-Browser Engines

The following browsers seem to work fine:

- Internet Explorer 5.5 (Windows version, if MSXML3 is installed)
- Internet Explorer 6 and newer
- Firefox 3.0 and newer
  - Be aware that XSLT execution can be suppressed using NoScript
  - Firefox does not load external DTDs nor external entities, see Mozilla Bug 22942, thus entities like &nbsp; need to be declared in the internal subset (Appendix C.1)
  - 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 \url{https://bugzilla.mozilla.org/show_bug.cgi?id=603159}).
  - Date computation is available in Firefox starting with Firefox 6 (see \url{https://bugzilla.mozilla.org/show_bug.cgi?id=603159}).
- Safari 3 (starting with version 3.0.4)
- Date computation not available (see \url{https://bugs.webkit.org/show_bug.cgi?id=4079})
- Google Chrome
  - Date computation not available (see \url{https://bugs.webkit.org/show_bug.cgi?id=4079})
- Opera (starting with version 10)
  - Date computation not available

The following browsers are known not to work properly:

- Firefox 1.*/2.*: (missing extension function - see change request at Mozilla BugZilla 193678)
- 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).

\textsuperscript{1} \url{http://www.exslt.org/exsl/functions/node-set/index.html}
\textsuperscript{2} \url{http://www.exslt.org/date/functions/date-time/}
\textsuperscript{3} \url{http://www.microsoft.com/downloads/details.aspx?FamilyID=2FB55371-C94E-4373-B0E9-DB4816552E41}
\textsuperscript{4} \url{https://addons.mozilla.org/de/firefox/addon/722}
\textsuperscript{5} \url{https://bugzilla.mozilla.org/show_bug.cgi?id=22942}
\textsuperscript{6} \url{http://bugzilla.mozilla.org/show_bug.cgi?id=193678}
• 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 10.4.4. Both problems have been reported through Apple's bug tracking system, see <http://drakken.dbc.mt view.ca.us/pipermail/xml2rfc/2005-May/002073.html> and <http://bugs.webkit.org/show_bug.cgi?id=407 9>.
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 the HTML 4.01 strict DTD [HTML]. This can be checked using the W3C's online validator at <http://validator.w3.org>.

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 Widgets7 extension.

The following LINK elements are produced:

<table>
<thead>
<tr>
<th>LINK type</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternate</td>
<td>for RFCs, a link to the authoritative ASCII version on the IETF web site</td>
</tr>
<tr>
<td>appendic</td>
<td>pointer to all top-level appendices</td>
</tr>
<tr>
<td>author</td>
<td>pointer to &quot;authors&quot; section</td>
</tr>
<tr>
<td>chapter</td>
<td>pointer to all top-level sections</td>
</tr>
<tr>
<td>contents</td>
<td>pointer to table of contents</td>
</tr>
<tr>
<td>copyright</td>
<td>pointer to copyright statement</td>
</tr>
<tr>
<td>index</td>
<td>pointer to index</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>META name</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>generator</td>
<td>from XSLT engine version and stylesheet version</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>META name</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC.Creator</td>
<td>from author information in front section</td>
</tr>
<tr>
<td>DC.Date.Issued</td>
<td>from date information in front section</td>
</tr>
<tr>
<td>DC.Description.Abstract</td>
<td>from abstract</td>
</tr>
<tr>
<td>DC.Identifier</td>
<td>document URN [RFC2648] from &quot;docName&quot; attribute</td>
</tr>
<tr>
<td>DC.isPartOf</td>
<td>RFC ISSN (for RFCs)</td>
</tr>
<tr>
<td>DC.Relation.Replaces</td>
<td>from &quot;obsoletes&quot; attribute</td>
</tr>
</tbody>
</table>
7. Transforming to XHTML

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

**Note:** Microsoft Internet Explorer does *not* support XHTML. Therefore it usually makes more sense to generate plain old HTML.
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 (xsl11toFop.xslt, xsl11toXep.xslt, xsl11toAn.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:

```bash
saxon rfc2616.xml rfc2629toFo.xslt > tmp.fo
saxon tmp.fo xsl11toFop.xslt > rfc2629.fo
```

9.2 Via X(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.

An alternative is PrinceXML (see <http://www.princexml.com/>), which can produce PDF directly from (X)HTML input, based on the CSS printing information.

For instance, PDF output with text justification turned on can be produced with:

```bash
saxon input.xml rfc2629toXHTML.xslt xml2rfc-ext-justification=print \
  > output.xhtml
prince output.xhtml 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).

11.3 <bcp14> element

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

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

```xml
<!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>">]>
```

11.4 <bb> element

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

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

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

11.6 <blockquote> element

This element is like the <blockquote> element defined in Section 9.2.2 of [HTML] (note this is a block-level element!). It should contain one or more <t> child elements.

11.7 <boilerplate> element

Can be used to include boilerplate (status, copyright, ...) into the front or back section. <section> elements within <x:boilerplate> appear as unnumbered sections in the output.

This element currently can not be "down-translated" for use in xml2rfc!

11.8 <bt> element

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

11.9 <dfn> element

This element is like the <dfn> element defined in Section 9.2.1 of [HTML].

11.10 <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&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.

11.11 <h> element

This element is like the "h9" element in [XHTML2].

11.12 <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.

11.13 <length-of> element

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

9 http://www.w3.org/TR/2006/WD-xhtml2-20060726/mod-structural.html#edef_structural_h
Example: computing the Content-Length header value

```xml
<artwork>
  ...
  Content-Length: <x:length-of target="req"/>
</artwork>
<x:span anchor="req">123456789</x:span>
<x:span><artwork/></x:span>
```

The length 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.

### 11.14 `<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>` element with identical attributes gets generated.

Example: generating HTML link element

```xml
<x:link xmlns:x="http://purl.org/net/xml2rfc/ext"
  rel="Bookmark"
  title="IETF WEBDAV Working Group"
</x:link>
```

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" or "prev" links in a series of documents. In this case, the href attribute is not required.

For instance:

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

### 11.15 `<lt>` element

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

### 11.16 `<note>` element

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

### 11.17 `<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).

### 11.18 `<prose>` element

10 http://www.w3.org/TR/html4/struct/links.html#edef-LINK
This element can be used inside <reference> to add plain text (before the date, when present).

See also <refcontent> (Section 12.13).

11.19  <q> element

This element is like the <q> element defined in Section 9.2.2 of [HTML].

11.20  <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

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

are equivalent to...

```
<xref target="test">test</xref>
<xref target="test">alias1</xref>
<xref target="test">alias 2</xref>
```

11.21  <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.

For example:

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

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

11.22  <sup> element

This element is like the <sup> element in Section 9.2.3 of [HTML].
Note: the down conversion to RFC2629 format replaces "xy" by "x^y".

See also <refcontent> (Section 12.18).

11.23 Extensions to Xml2rfc <artwork> element

Sometimes, artwork occurs inside lists. To get it indent 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".

11.24 Extensions to Xml2rfc <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.

11.25 Extensions to Xml2rfc <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 \textit{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.

11.26 Extensions to Xml2rfc <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]).

11.27 Extensions to Xml2rfc <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.

11.28 Extensions to Xml2rfc <xref> element

Three 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.

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

- (Comma) [reference], Section sec
- () [reference] (Section sec)
of  Section sec of [reference]

number  sec

none    No output (can be used to have xrefs to references without having them rendered as such)

sec     Section sec

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 XML2RFCv3 Vocabulary

rfc2629.xslt experimentally supports some elements from the "V3" vocabulary, defined in [XML2RFCV3]. This support is experimental, as the "v3" vocabulary is still being developed.

12.1 aside Element

See Section 2.6 of [XML2RFCV3].

12.2 bcp14 Element

See Section 2.9 of [XML2RFCV3].

12.3 blockquote Element

See Section 2.10 of [XML2RFCV3].

12.4 displayreference Element

See Section 2.19 of [XML2RFCV3].

12.5 dd Element

See Section 2.18 of [XML2RFCV3].

12.6 dl Element

See Section 2.20 of [XML2RFCV3].

12.7 dt Element

See Section 2.21 of [XML2RFCV3].

12.8 em Element

See Section 2.22 of [XML2RFCV3].

12.9 li Element

See Section 2.29 of [XML2RFCV3].

12.10 name Element

See Section 2.32 of [XML2RFCV3]. Currently only supported inside <references> and <section>.

12.11 ol Element

See Section 2.34 of [XML2RFCV3].

12.12 postalLine Element

See Section 2.38 of [XML2RFCV3].
12.13  refcontent Element

See Section 2.39 of [XML2RFCV3].

12.14  Extensions to reference Element

See Section 2.40 of [XML2RFCV3].

12.14.1  quoteTitle attribute

See Section 2.40.2 of [XML2RFCV3].

12.15  Extensions to section Element

See Section 2.45 of [XML2RFCV3].

12.15.1  numbered attribute

See Section 2.45.2 of [XML2RFCV3].

12.15.2  removeInRFC attribute

See Section 2.45.4 of [XML2RFCV3].

12.16  strong Element

See Section 2.49 of [XML2RFCV3].

12.17  sub Element

See Section 2.50 of [XML2RFCV3].

12.18  sup Element

See Section 2.51 of [XML2RFCV3].

12.19  tt Element

See Section 2.61 of [XML2RFCV3].

12.20  ul Element

See Section 2.62 of [XML2RFCV3].

12.21  Extensions to xref Element

12.21.1  relative attribute

See Section 2.65.3 of [XML2RFCV3].

12.21.2  section attribute

See Section 2.65.4 of [XML2RFCV3].
12.21.3 sectionFormat attribute

See Section 2.65.5 of [XML2RFCV3].
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 <ftp://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 11.26).

**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 <http://greenbytes.de/tech/webdav/rfc2518.xml>:

```bash
> 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 "W3C ", or must start with "World Wide Web Consortium ", 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/](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
<?xml version="1.0" encoding="utf-8"?>
<references>
  <reference target="urn:isbn:0134516591">
    <front>
      <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 RFC2629bis DTD

clean-for-DTD.xslt can be used to down-convert some extensions to a format that is supported by the base xml2rfc distribution. 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:bc>`, `<x:bb>` 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.
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:

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

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

```bash
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:

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

13.6 GRDDL

`rfc2629grddl.xslt` extracts RDF information. This is experimental work-in-progress. See [http://www.w3.org/TR/grddl/](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:

1. Mozilla Firefox, 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:

```xml
<?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
<?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.
14. Informative References


A. RELAX NG Compact Schema

The RelaxNG schema ([RNC]) below can be used to validate input documents (for instance, with Jing\(^1\)).

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

---

\(^1\) [http://www.thaiopensource.com/relaxng/jing.html](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#"

# Include RFC2629bis RNC grammar
include "rfc2629.rnc"
{
    # 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 {
                (TEXT
                    | eref
                    | iref
                    | spanx
                    | xref
                    | v3_em
                    | v3_strong
                    | x_abnf-char-sequence
                    | x_bb
                    | x_bc
                    | x_bcpl4
                    | x_bt
                    | x_highlight
                    | x_length-of
                    | x_parse-xml
                    | x_ref
                    | x_span
                    | x_x)*
            } 
        }

    # Redefine <back> to allow boilerplate
    back =
        element back {
            attlist back {
                v3_displayreference*, references*, section*,
                x_boilerplate?
            }
        }
}
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:

<table>
<thead>
<tr>
<th>Type</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>abnf</td>
<td>ABNF as per [RFC5234]</td>
</tr>
<tr>
<td>abnf2045</td>
<td>ABNF as per [RFC2045]</td>
</tr>
<tr>
<td>abnf2616</td>
<td>ABNF as per [RFC2616], Section 2.1</td>
</tr>
<tr>
<td>application/relax-ng-compact-syntax</td>
<td>Relax NG Compact Syntax as per [RNC]</td>
</tr>
<tr>
<td>application/xml-dtd</td>
<td>XML DTD</td>
</tr>
<tr>
<td>code</td>
<td>monospaced text (with outline)</td>
</tr>
<tr>
<td>drawing</td>
<td>drawing (with outline)</td>
</tr>
<tr>
<td>example</td>
<td>monospaced text (with outline)</td>
</tr>
<tr>
<td>inline</td>
<td>monospaced text (no outline)</td>
</tr>
<tr>
<td>message/http; msgtype=&quot;request&quot;</td>
<td>HTTP message, as per [RFC2616], Section 19.1</td>
</tr>
<tr>
<td>message/http; msgtype=&quot;response&quot;</td>
<td>HTTP message, as per [RFC2616], Section 19.1</td>
</tr>
</tbody>
</table>
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
<?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 11.3 -->
  <!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 -->
  <!-- allow a custom reference using "&mydraft;" -->
  <!-- the data will be fetched from the same location as the source file -->
  <!ENTITY mydraft PUBLIC "reference.mydraft.xml" ]>
```

Note: including entities from a remote site will not work in Firefox, see <https://bugzilla.mozilla.org/show_bug.cgi?id=22942>.

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.

```xml
<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.
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 RFC2629-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.

D.1 The `/rfc/@ipr` Attribute

As of the time of this writing, this attribute value can take a long list of values. As frequently, this is not the result of a grand plan, but simply for historic reasons. Of these values, only a few are currently in use; all others are supported by the various tools for backwards compatibility with old source files.

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

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

Finally, for the current "Status Of This Memo" text, the `submissionType` attribute determines whether a statement about "Code Components" is inserted (this is the case for the value "IETF", which also happens to be the default). Other values, such as "independent", suppress this part of the text.

D.1.1 Current Values: '*trust200902'

The name for these values refers to the "TLP" ("IETF TRUST Legal Provisions Relating to IETF Documents"), on effect February 15, 2009 (see <http://trustee.ietf.org/license-info/archive/IETF-Trust-License-Policy-20090215.pdf>). Updates to this document were published on September 12, 2009 (TLP 3.0, <http://trustee.ietf.org/license-info/archive/IETF-Trust-License-Policy-20090912.pdf>) and on December 28, 2009 (TLP 4.0, <http://trustee.ietf.org/license-info/archive/IETF-Trust-License-Policy-20091228.pdf>), modifying the license for code components. The actual text is located in Section 6 ("Text To Be Included in IETF Documents") of these documents.

The tools will automatically produce the "right" text depending on the document's date information (see above):

<table>
<thead>
<tr>
<th>TLP</th>
<th>URI</th>
<th>starting with publication date</th>
</tr>
</thead>
</table>

D.1.1.1 trust200902

This should be the default, unless one of the more specific '*trust200902' values is a better fit. It produces the text in Sections 6.a and 6.b of the TLP.
D.1.1.2  noModificationTrust200902

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

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

Note: this clause is incompatible with RFCs to be published on the Standards Track.

D.1.1.3  noDerivativesTrust200902

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

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

Note: this clause is incompatible with RFCs.

D.1.1.4  pre5378Trust200902

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

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

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

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

D.1.2  Historic Values

D.1.2.1  Historic Values: '*trust200811'

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

D.1.2.2  Historic Values: '*3978'

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

D.1.2.3  Historic Values: '*3667'

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

D.1.2.4  Historic Values: '*2026'

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

The special value "none" was also used back then, and denied the IETF any rights beyond publication as Internet Draft.
D.2 The /rfc/@category Attribute

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

For Internet Drafts, the category attribute is not needed, but will appear on the front page ("Intended Status"). Supplying this information can be useful, because reviewers may want to know.

Note: the Standards Track consists of "Proposed Standard", "Draft Standards", and "Internet Standard". These do not appear in the boilerplate, thus the category attribute doesn't handle them. However, this information can be useful for validity checkers, and thus rfc2629.xslt supports an extension attribute for that purpose (see Section 11.26 for details).

D.3 The /rfc/@submissionType Attribute

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

Not surprisingly, the values for the attribute are "IETF" (the default value), "independent", "IAB", and "IRTF". Historically, this did not affect the final appearance of RFCs, except for subtle differences in Copyright notices.

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

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

D.4 The /rfc/@consensus Attribute

For some of the publication streams (see Appendix D.3), the "Status Of This Memo" section depends on whether there was a consensus to publish (again, see Section 3.2.2 of [RFC5741]).

The consensus attribute ("yes"/"no", defaulting to "yes") can be used to supply this information. The effect for the various streams is:
- "independent" and "IAB": none.
- "IETF": mention that there was an IETF consensus.
- "IRTF": mention that there was a research group consensus (where the name of the research group is extracted from the <workgroup> element).

D.5 The /rfc/@number Attribute

For RFCs, this attribute supplies the RFC number.

D.6 The /rfc/@docName Attribute

For Internet Drafts, this specifies the draft name (which appears below the title). The file extension is not part of the draft, so in general it should end with the current draft number ("-", plus two digits).

Note: "Idnits" (<http://tools.ietf.org/tools/idnits/> ) checks the in-document draft name for consistency with the filename of the submitted document.
D.7 The /rfc/@obsoletes Attribute

The RFC Editor maintains a database (http://www.rfc-editor.org/rfc.html) of all RFCs, including information about which one obsoletes which. Upon publication of an RFC, this database is updated from the data on the front page.

This attribute takes a list of comma-separated RFC numbers. Do not put the string "RFC" here.

D.8 The /rfc/@updates Attribute

This is like obsoletes, but for the "updates" relation.
E. License

Copyright (c) 2006-2014, 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 19, 37
  allow-markup-in-artwork PI pseudo-attribute 8
  alternate HTML LINK element 13
  anchor-alias Extension Element 19, 37
  Anchors
    rfc.abstract 10
    rfc.authors 10
    rfc.copyright 10
    rfc.copyrightnotice 10
    rfc.figure.n 10, 10
    rfc.figure.u.n 10, 10
    rfc.index 10
    rfc.ipr 10
    rfc.iref.n 10
    rfc.references 10, 10
    rfc.section.n 10
    rfc.section.n.p.m 10
    rfc.status 10
    rfc.toc 10
    rfc.xref.name.n 10
  AntennaHouse XSL Formatter 17
  Apache FOP 17
  appendix HTML LINK element 13
  aside V3 Extension Element 25, 37
  assign-section-number Extension Element 37
  author HTML LINK element 13
  authors-section PI pseudo-attribute 8

B
  background PI pseudo-attribute 7
  bb Extension Element 19, 37
  bc Extension Element 20, 37
  bcp14 Extension Element 19, 37
  bcp14 V3 Extension Element 25, 37
  BCP97 28, 34
  blockquote Extension Element 20, 37
  blockquote V3 Extension Element 25, 37
  boilerplate Extension Element 20, 37
  bt Extension Element 20, 37

C
  chapter HTML LINK element 13
  CHM format 16
  comments PI pseudo-attribute 7
  compact PI pseudo-attribute 7
  contents HTML LINK element 13
  copyright HTML LINK element 13
  Creator DCMI property 14

D
  Date.Issued DCMI property 14
  DC-HTML 14, 34
  DCMI properties
    Creator 14
    Date.Issued 14
    Description.Abstract 14
    Identifier 14
    isPartOf 14
    Relation.Replaces 14
  dd V3 Extension Element 25, 37
  Description.Abstract DCMI property 14
  dfn Extension Element 20, 37
  displayreference V3 Extension Element 25, 37
  dl V3 Extension Element 25, 37
  dt V3 Extension Element 25, 37
duplex PI pseudo-attribute 8

E
  editing PI pseudo-attribute 7
  em V3 Extension Element 25, 37
  Extension Elements
    abnf-char-sequence 19, 37
    anchor-alias 19, 37
    assign-section-number 37
    bb 19, 37
    bc 20, 37
    bcp14 19, 37
    blockquote 20, 37
    boilerplate 20, 37
    bt 20, 37
    dfn 20, 37
    feedback 20, 37
    h 20, 37
    highlight 20, 37
    include-author 37
    length-of 20, 37
    link 21, 37
    lt 21, 37
    note 21, 37
    parse-xml 21, 37
    prose 21, 37
    q 22, 37
    rdf:Description 37
    ref 22, 37
    source 22, 37
    span 37
    sup 22, 37
    x 37

F
  feedback Extension Element 20, 37
  Firefox
    1.*/2.* 11
    3.* 11
  footer PI pseudo-attribute 7

G
  generator HTML META element 13
  Google Chrome 11
  Grammar 36

H
  h Extension Element 20, 37
  header PI pseudo-attribute 7
  highlight Extension Element 20, 37
  HTML 13, 20, 22, 22, 34
Section 9.2.1 20
Section 9.2.2 20, 22
Section 9.2.3 22
HTML compliance 13
HTML LINK elements
alternate 13
appendix 13
author 13
chapter 13
contents 13
copyright 13
index 13
HTML META elements
generator 13
keywords 13

I
Identifier DCMI property 14
include PI pseudo-attribute 8
include-author Extension Element 37
include-index PI pseudo-attribute 8
include-references-in-index PI pseudo-attribute 8
index HTML LINK element 13
inline PI pseudo-attribute 7
insert-metadata PI pseudo-attribute 8
Internet Explorer 5.5 11
Internet Explorer 6 11
Internet Explorer 7 11
Internet Explorer 8 11
Internet Explorer 9 11
iprnotified PI pseudo-attribute 7
isPartOf DCMI property 14

J
justification PI pseudo-attribute 8

K
keywords HTML META element 13

L
length-of Extension Element 20, 37
li V3 Extension Element 25, 37
link Extension Element 21, 37
linkmailto PI pseudo-attribute 7
lt Extension Element 21, 37

M
Microsoft Help 16
Mozilla 11, 11
MSXML3 11
MSXML4 11

N
name V3 Extension Element 25, 37
needLines PI pseudo-attribute 8
NoScript 11
note Extension Element 21, 37

O
ol V3 Extension Element 25, 37
Opera 11, 11
subcompact 8
support-rfc2731 8
symrefs 7
toc 7
tocdepth 7
tocindent 7
tocompact 7
prose Extension Element 21, 37

Q
q Extension Element 22, 37

R
rdf:Description Extension Element 37
ref Extension Element 22, 37
refcontent V3 Extension Element 26, 37
reference V3 Extension Element 26
reffn:parent PI pseudo-attribute 7
Relation:Replaces DCMI property 14
RELAX NG Compact Schema 36
rfc.abstract anchor 10
rfc.authors anchor 10
rfc.copyright anchor 10
rfc.copyrightnotice anchor 10
rfc.figure.n anchor 10, 10
rfc.figure.u.n anchor 10, 10
rfc.index anchor 10
rfc.ipr anchor 10
rfc.iref.n anchor 10
rfc.note.n anchor 10
rfc.references anchor 10
rfc.references.n anchor 10
rfc.sectionanchor 10
rfc.section.n anchor 10
rfc.section.n.p.m anchor 10
rfc.status anchor 10
rfc.toc anchor 10
rfc.xref.name.n anchor 10
RFC2026 23, 34, 43
Section 4 43
Section 4.1 23
RFC204534, 38
RFC2119 19, 24, 34
Section 5 24
RFC2616 34, 38, 38, 38
Section 2.1 38
Section 19.1 38, 38
RFC2629S, 34
RFC264814, 34
RFC27314, 34
RFC273434, 38
RFC5741 34, 43, 43, 43
Section 3.2.2 43, 43
rfcedstyle PI pseudo-attribute 7
RNC34, 36, 38

S
Safari 11, 12
3.1 11
Saxon 11

Schema 36
sec-no-trailing-dots PI pseudo-attribute 8
section V3 Extension Element 26
slides PI pseudo-attribute 8
sortrefs PI pseudo-attribute 7
source Extension Element 22, 37
span Extension Element 37
strict PI pseudo-attribute 8
strong V3 Extension Element 26, 37
sub V3 Extension Element 26, 37
subcompact PI pseudo-attribute 8
sup Extension Element 22, 37
sup V3 Extension Element 26, 37
support-rfc2731 PI pseudo-attribute 8
symrefs PI pseudo-attribute 7

T
toc PI pseudo-attribute 7
tocdepth PI pseudo-attribute 7
tocindent PI pseudo-attribute 8
tocompact PI pseudo-attribute 8
topblock PI pseudo-attribute 7
tt V3 Extension Element 26, 37

U
ul V3 Extension Element 26, 37

V
V3 Extension Elements
aside 25, 37
bcp14 25, 37
blockquote 25, 37
dd 25, 37
displayreference 25, 37
dl 25, 37
dt 25, 37
dm 25, 37
li 25, 37
name 25, 37
ol 25, 37
postallLine 25, 37
reffn:content 26, 37
reference 26
section 26
strong 26, 37
sub 26, 37
sup 26, 37
tt 26, 37
ul 26, 37

X
x Extension Element 37
Xalan 11
XHTML2 20, 34
XML 34, 39
Section 2.8 39
xml-stylesheet PI 13
xml2rfc-background parameter 7
xml2rfc-comments parameter 7
xml2rfc-editing parameter 7, 7
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>xml2rfc-ext-allow-markup-in-artwork</td>
<td>2.625</td>
</tr>
<tr>
<td>xml2rfc-ext-authors-section</td>
<td>2.1025</td>
</tr>
<tr>
<td>xml2rfc-ext-duplex</td>
<td>2.1825</td>
</tr>
<tr>
<td>xml2rfc-ext-include-index</td>
<td>2.1925</td>
</tr>
<tr>
<td>xml2rfc-ext-include-references-in-index</td>
<td>2.2025</td>
</tr>
<tr>
<td>xml2rfc-ext-insert-metadata</td>
<td>2.2125</td>
</tr>
<tr>
<td>xml2rfc-ext-justification</td>
<td>2.2225</td>
</tr>
<tr>
<td>xml2rfc-ext-parse-xml-in-artwork</td>
<td>2.2925</td>
</tr>
<tr>
<td>xml2rfc-ext-sec-no-trailing-dots</td>
<td>2.3225</td>
</tr>
<tr>
<td>xml2rfc-ext-support-rfc2731</td>
<td>2.3425</td>
</tr>
<tr>
<td>xml2rfc-footnotes</td>
<td>2.3825</td>
</tr>
<tr>
<td>xml2rfc-header</td>
<td>2.3926</td>
</tr>
<tr>
<td>xml2rfc-inline</td>
<td>2.4026</td>
</tr>
<tr>
<td>xml2rfc-iprnotified</td>
<td>2.40.226</td>
</tr>
<tr>
<td>xml2rfc-linkmailto</td>
<td>2.45.26</td>
</tr>
<tr>
<td>xml2rfc-private</td>
<td>2.45.26</td>
</tr>
<tr>
<td>xml2rfc-refparent</td>
<td>2.45.26</td>
</tr>
<tr>
<td>xml2rfc-rfcedstyle</td>
<td>2.45.26</td>
</tr>
<tr>
<td>xml2rfc-sortrefs</td>
<td>2.4926</td>
</tr>
<tr>
<td>xml2rfc-symrefs</td>
<td>2.5026</td>
</tr>
<tr>
<td>xml2rfc-toc</td>
<td>2.5126</td>
</tr>
<tr>
<td>xml2rfc-tocdepth</td>
<td>2.6126</td>
</tr>
<tr>
<td>xml2rfc-topblock</td>
<td>2.6226</td>
</tr>
<tr>
<td>XML2RFCV2</td>
<td>2.65.326</td>
</tr>
<tr>
<td>XML2RFCV3</td>
<td>2.65.426</td>
</tr>
<tr>
<td>XSL-FO</td>
<td>2.65.527</td>
</tr>
<tr>
<td>xsltproc</td>
<td>17</td>
</tr>
<tr>
<td>passing parameters</td>
<td>35</td>
</tr>
</tbody>
</table>
Author's Address

Julian F. Reschke
greenbytes GmbH
Hafenweg 16
Muenster, NW 48155
Germany
EMail: julian.reschke@greenbytes.de
URI: http://greenbytes.de/tech/webdav/