

# **Transforming RFC2629-formatted XML through XSLT**

## Table of Contents

<b>1 Introduction.....</b>	<b>3</b>
<b>2 Supported RFC2629 elements.....</b>	<b>4</b>
2.1 Extension elements.....	4
<b>3 Processing Instructions.....</b>	<b>5</b>
3.1 Supported xml2rfc-compatible PIs.....	5
3.2 Unsupported xml2rfc-compatible PIs.....	5
3.3 Extension PIs.....	5
<b>4 Anchors.....</b>	<b>7</b>
<b>5 Supported XSLT engines.....</b>	<b>8</b>
5.1 Standalone Engines.....	8
5.2 In-Browser Engines.....	8
<b>6 Transforming to HTML.....</b>	<b>9</b>
6.1 HTML compliance.....	9
6.2 Standard HTML LINK elements.....	9
6.3 Standard HTML metadata.....	9
6.4 Dublin Core (RFC2731) metadata.....	10
6.5 Experimental hCard support.....	10
<b>7 Transforming to XHTML.....</b>	<b>11</b>
<b>8 Transforming to CHM (Microsoft Compiled Help).....</b>	<b>12</b>
<b>9 Transforming to PDF via XSL-FO.....</b>	<b>13</b>
9.1 Extension feature matrix.....	13
9.2 Example: producing output for Apache FOP.....	13
<b>10 Utilities.....</b>	<b>15</b>
10.1 Checking References.....	15
10.2 Producing reference entries for books.....	15
10.3 Down-converting to RFC2629bis DTD.....	15
10.4 Extracting artwork.....	16
<b>11 Informative References.....</b>	<b>17</b>
<b>Author's Address.....</b>	<b>18</b>
<b>Index.....</b>	<b>19</b>

## 1. Introduction

This document describes a set of XSLT transformations that can be used to transform RFC2629-compliant XML (see [\[RFC2629\]](#)) 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 through XSL-FO ([Section 9](#)) and
- various utilities ([Section 10](#)).

## 2. Supported RFC2629 elements

`rfc2629.xslt` supports both all RFC2629 grammar elements and the extensions implemented in `xml2rfc 1.21`.

### 2.1 Extension elements

In addition, `rfc2629.xslt` supports a set of extension elements, using elements and attributes in the namespace "`http://greenbytes.de/2002/rfcedit`". They are used for

- simple issue tracking and change tracking and
- adding additional metadata to the generated documents (such as HTML LINK elements to related documents, see [Section 6.2](#)).

Note that these extensions are experimental. 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 foo.xml rfc2629.xslt xml2rfc-toc=yes \
  xml2rfc-ext-support-rfc2731=no > result.hzml
```

#### 3.1 Supported xml2rfc-compatible PIs

PI target	PI pseudo-attribute	XSLT parameter name	default	comment
rfc	background	xml2rfc-background	(not set)	
rfc	compact	xml2rfc-compact	"no"	only applies to HTML output method when printing
rfc	comments	xml2rfc-comments	(not set)	
rfc	editing	xml2rfc-editing	"no"	
rfc	footer	xml2rfc-footer	(not set)	
rfc	header	xml2rfc-header	(not set)	
rfc	inline	xml2rfc-inline	(not set)	
rfc	iprnotified	xml2rfc-iprnotified	"no"	
rfc	linkmailto	xml2rfc-linkmailto	"yes"	
rfc	private	xml2rfc-private	(not set)	
rfc	sortrefs	xml2rfc-sortrefs	"no"	
rfc	symrefs	xml2rfc-symrefs	"no"	
rfc	toc	xml2rfc-toc	"no"	
rfc	tocdepth	xml2rfc-tocdepth	99	
rfc	topblock	xml2rfc-topblock	"yes"	

#### 3.2 Unsupported xml2rfc-compatible PIs

PI target	PI pseudo-attribute	comment
rfc	include	incompatible with XML/XSLT processing model
rfc	needLines	
rfc	slides	
rfc	strict	
rfc	subcompact	
rfc	tocindent	(defaults to "yes")
rfc	tocompact	

#### 3.3 Extension PIs

PI target	PI pseudo-attribute	XSLT parameter name	default	description
rfc-ext	allow-markup-in-artwork	xml2rfc-allow-markup-in-artwork	no	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 <a href="#">Section 10.3</a> ).
rfc-ext	authors-section	xml2rfc-ext-authors-section	no	When "end", place the authors section at the end (just before the copyright statements). This seems to be the preferred order in the newest RFCs.
rfc-ext	parse-xml-in-artwork	xml2rfc-parse-xml-in-artwork	no	May be used to enable parsing of XML content in figures (MSXML only).
rfc-ext	support-rfc2731	xml2rfc-ext-support-rfc2731	no	Decides whether the HTML transformation should generate META tags according <a href="#">Section 6.4</a> .
rfc-ext	sec-no-trailing-dots	xml2rfc-ext-sec-no-trailing-dots	no	When set to "yes", add trailing dots to section numbers. This seems to be the preferred format in the newest RFCs.

## 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

## 5. Supported XSLT engines

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

### 5.1 Standalone Engines

The following XSLT engines are believed to work well:

- MSXML3 and MSXML4 (<<http://msdn.microsoft.com/xml>>; these processors do not support exsl:node-set(), but have a similar proprietary extension)
- Saxon (<<http://saxon.sourceforge.net/>>)
- Xalan (<<http://xml.apache.org/xalan-j/>>)
- xsltproc (libxslt) (<<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 (Windows version)

The following browsers are known not to work properly:

- Mozilla (missing extension function - see change request at Mozilla bugs [193678](http://bugzilla.mozilla.org/show_bug.cgi?id=193678)<sup>2</sup> and [215242](http://bugzilla.mozilla.org/show_bug.cgi?id=215242)<sup>3</sup>)
- Safari supports client-side XSLT as of MacOS X 10.4, but misses required extension functions. There also seems to be a problem with stylesheets producing non-ASCII output (such as NBSP characters). Both problems have been reported through Apple's bug tracking system.

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

<sup>2</sup> [http://bugzilla.mozilla.org/show\\_bug.cgi?id=193678](http://bugzilla.mozilla.org/show_bug.cgi?id=193678)

<sup>3</sup> [http://bugzilla.mozilla.org/show\\_bug.cgi?id=215242](http://bugzilla.mozilla.org/show_bug.cgi?id=215242)



## 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>.

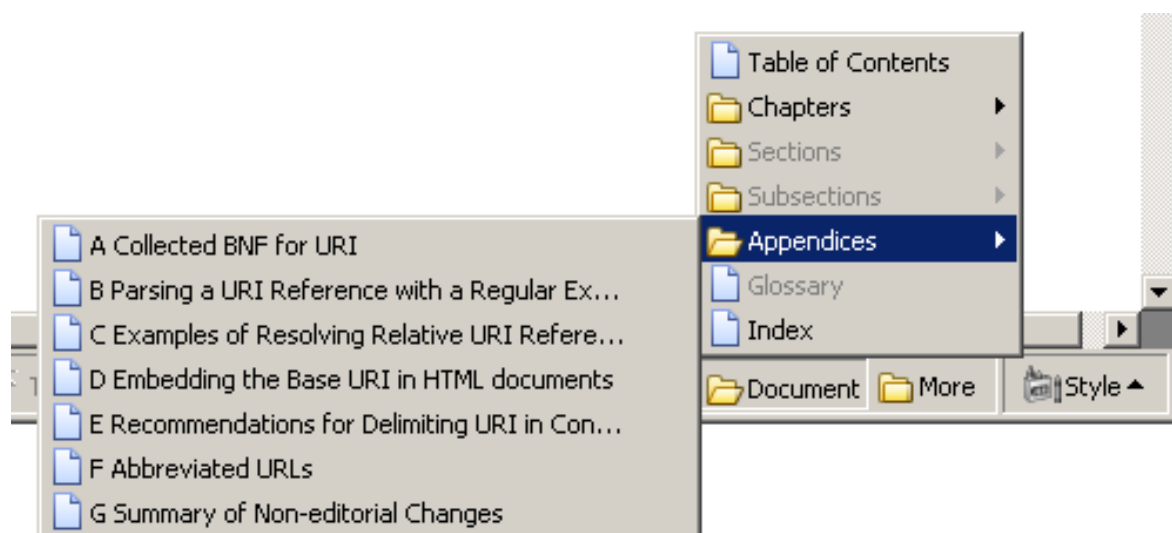
### 6.2 Standard HTML LINK elements

LINK elements exist since HTML 2.0. They can be used to embed content-independant links inside the document. Unfortunately, only few user agents fully support this element, namely Mozilla where it's called "Site Navigation Bar" (by default disabled!).

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 appendices
author	pointer to "authors" section
chapter	pointer to all top-level sections
contents	pointer to table of contents
copyright	pointer to copyright statement
index	pointer to index

The figure below shows how Mozilla Firefox 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 [\[RFC2731\]](#).

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 <a href="#">[RFC2648]</a> from "docName" attribute
DC.Relation.Replaces	from "obsoletes" attribute

## 6.5 Experimental hCard support

The generated author information is formatted in [hCard](#)<sup>4</sup> format.

<sup>4</sup> <http://developers.technorati.com/wiki/hCard>

## 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 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 (page breaking, table layout), and some popular extensions aren't standardized yet, the translation produces a generic output (hopefully) conforming to [XSL-FO-11-WD]. Specific backends (`xsl11toFop.xslt`, `xsl11toXep.xslt`, `xsl11toAn.xslt`) the provide post-processing for the individual processors.

### 9.1 Extension feature matrix

	PDF anchors	PDF bookmarks	PDF document information	Index cleanup
<a href="#">XSL 1.1 WD</a> <sup>5</sup>	no, but can be auto-generated from "id" attributes	<a href="#">yes</a> <sup>6</sup>	no, but uses XEP output extensions	<a href="#">yes</a> <sup>7</sup>
<a href="#">Antenna House XSL formatter</a> <sup>8</sup>	no	<a href="#">yes</a> <sup>9</sup> (from XSL 1.1 bookmarks)	<a href="#">yes</a> <sup>10</sup> (from XEP document info)	<a href="#">yes</a> <sup>11</sup> (just page duplicate elimination, from XSL 1.1 page index)
<a href="#">Apache FOP</a> <sup>12</sup>	<a href="#">yes</a> <sup>13</sup>	<a href="#">yes</a> <sup>14</sup> (from XSL 1.1 bookmarks)	no	no
<a href="#">RenderX XEP</a> <sup>15</sup>	no	<a href="#">yes</a> <sup>16</sup> (from XSL 1.1 bookmarks)	<a href="#">yes</a> <sup>17</sup>	<a href="#">yes</a> <sup>18</sup> (from XSL 1.1 page index)

### 9.2 Example: producing output for Apache FOP

Example:

<sup>5</sup> <http://www.w3.org/TR/2003/WD-xsl11-20031217/>

<sup>6</sup> <http://www.w3.org/TR/2003/WD-xsl11-20031217/#d0e12873>

<sup>7</sup> <http://www.w3.org/TR/2003/WD-xsl11-20031217/#d0e12534>

<sup>8</sup> <http://www.antennahouse.com/>

<sup>9</sup> <http://www.antennahouse.com/XSL20/axf-extension.htm>

<sup>10</sup> <http://www.antennahouse.com/XSL20/axf-extension.htm>

<sup>11</sup> <http://www.antennahouse.com/XSL20/axf-extension.htm>

<sup>12</sup> <http://xml.apache.org/fop/>

<sup>13</sup> <http://xml.apache.org/fop/extensions.html#named-destinations>

<sup>14</sup> <http://xml.apache.org/fop/extensions.html#bookmarks>

<sup>15</sup> <http://xep.xattic.com/>

<sup>16</sup> <http://xep.xattic.com/xep/spec.html>

<sup>17</sup> <http://xep.xattic.com/xep/spec.html>

<sup>18</sup> <http://xep.xattic.com/xep/spec.html>

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

## 10. Utilities

### 10.1 Checking References

`check-ietf-references.xslt` can be used to check all references to RFC-series IETF publications (note this script requires a local copy of <http://ftp.isi.edu/in-notes/rfc-index.xml>). For instance:

```
> saxon rfc2518.xml check-ietf-references.xslt
Normative References:
RFC1766: [PROPOSED STANDARD] obsoleted by RFC3066 RFC3282
RFC2277: [BEST CURRENT PRACTICE] (-> BCP0018) ok
RFC2119: [BEST CURRENT PRACTICE] (-> BCP0014) ok
RFC2396: [DRAFT STANDARD] ok
RFC2069: [PROPOSED STANDARD] obsoleted by RFC2617
RFC2068: [PROPOSED STANDARD] obsoleted by RFC2616
RFC2141: [PROPOSED STANDARD] ok
RFC2279: [PROPOSED STANDARD] obsoleted by RFC3629
Informational References:
RFC2026: [BEST CURRENT PRACTICE] (-> BCP0009) ok
RFC1807: [INFORMATIONAL] ok
RFC2291: [INFORMATIONAL] ok
RFC2413: [INFORMATIONAL] ok
RFC2376: [INFORMATIONAL] obsoleted by RFC3023
```

### 10.2 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...

### 10.3 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:

- `<ieref>` 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 marlup is stripped, but the element is replaced by the applicable replacement text).

## 10.4 Extracting artwork

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

For instance:

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



## 11 Informative References

- [RFC2629] Rose, M.T., "[Writing I-Ds and RFCs using XML](#)", RFC 2629, June 1999.
- [RFC2648] Moats, R., "[A URN Namespace for IETF Documents](#)", RFC 2648, August 1999.
- [RFC2731] Kunze, J.A., "[Encoding Dublin Core Metadata in HTML](#)", RFC 2731, December 1999.
- [HTML] Raggett, D., Hors, A., and I. Jacobs, "[HTML 4.01 Specification](#)", <<http://www.w3.org/TR/html401/>>, W3C REC REC-html401-19991224, December 1999.
- [XSL-FO] Adler, S., Berglund, A., Caruso, J., Deach, S., Graham, T., Grosso, P., Gutentag, E., Milowski, R., Parnell, S., Richman, J., and S. Zilles, "[Extensible Stylesheet Language \(XSL\) Version 1.0](#)", <<http://www.w3.org/TR/2001/REC-xsl-20011015/>>, W3C REC REC-xsl-20011015, October 2001.
- [XSL-FO-11-WD] Berglund, A., "[Extensible Stylesheet Language \(XSL\) Version 1.1](#)", <<http://www.w3.org/TR/2004/WD-xsl11-20041216/>>, W3C REC WD-xsl11-20041216, December 2004.

## Author's Address

**Julian F. Reschke**

greenbytes GmbH

Salzmannstrasse 152

Muenster, NW 48159

Germany

Phone: [+49 251 2807760](tel:+492512807760)Fax: [+49 251 2807761](tel:+492512807761)EMail: [julian.reschke@greenbytes.de](mailto:julian.reschke@greenbytes.de)URI: <http://greenbytes.de/tech/webdav/>

## Index

### A

- allow-markup-in-artwork PI pseudo-attribute [6](#)
- alternate HTML LINK element [9](#)
- Anchor
  - rfc.abstract [7](#)
  - rfc.authors [7](#)
  - rfc.copyright [7](#)
  - rfc.copyrightnotice [7](#)
  - rfc.figure.n [7](#), [7](#)
  - rfc.figure.u.n [7](#), [7](#)
  - rfc.index [7](#)
  - rfc.ipr [7](#)
  - rfc.iref.n [7](#)
  - rfc.note.n [7](#)
  - rfc.references [7](#), [7](#)
  - rfc.section.n [7](#)
  - rfc.section.n.p.m [7](#)
  - rfc.status [7](#)
  - rfc.toc [7](#)
- AntennaHouse XSL Formatter [13](#)
- Apache FOP [13](#)
- appendix HTML LINK element [9](#)
- author HTML LINK element [9](#)
- authors-section PI pseudo-attribute [6](#)

### B

- background PI pseudo-attribute [5](#)

### C

- chapter HTML LINK element [9](#)
- CHM format [12](#)
- comments PI pseudo-attribute [5](#)
- compact PI pseudo-attribute [5](#)
- contents HTML LINK element [9](#)
- copyright HTML LINK element [9](#)
- Creator DCMI property [10](#)

### D

- Date.Issued DCMI property [10](#)
- DCMI properties
  - Creator [10](#)
  - Date.Issued [10](#)
  - Description.Abstract [10](#)
  - Identifier [10](#)
  - Relation.Replaces [10](#)
- Description.Abstract DCMI property [10](#)

### E

- editing PI pseudo-attribute [5](#)

### F

- footer PI pseudo-attribute [5](#)

### G

- generator HTML META element [10](#)

### H

- header PI pseudo-attribute [5](#)
- HTML compliance [9](#)
- HTML LINK elements
  - alternate [9](#)
  - appendix [9](#)

- author [9](#)
- chapter [9](#)
- contents [9](#)
- copyright [9](#)
- index [9](#)

#### HTML META elements

- generator [10](#)
- keywords [10](#)

### I

- Identifier DCMI property [10](#)
- include PI pseudo-attribute [5](#)
- index HTML LINK element [9](#)
- inline PI pseudo-attribute [5](#)
- Internet Explorer 5.5 [8](#)
- Internet Explorer 6 [8](#)
- iprnotified PI pseudo-attribute [5](#)

### K

- keywords HTML META element [10](#)

### L

- linkmailto PI pseudo-attribute [5](#)

### M

- Microsoft Help [12](#)
- Mozilla [8](#)
- MSXML3 [8](#)
- MSXML4 [8](#)

### N

- needLines PI pseudo-attribute [5](#)

### P

#### Parameters

- xml2rfc-background [5](#)
- xml2rfc-comments [5](#)
- xml2rfc-compact [5](#)
- xml2rfc-editing [5](#)
- xml2rfc-ext-allow-markup-in-artwork [6](#)
- xml2rfc-ext-authors-section [6](#)
- xml2rfc-ext-parse-xml-in-artwork [6](#)
- xml2rfc-ext-sec-no-trailing-dots [6](#)
- xml2rfc-ext-support-rfc2731 [6](#)
- xml2rfc-footer [5](#)
- xml2rfc-header [5](#)
- xml2rfc-inline [5](#)
- xml2rfc-iprnotified [5](#)
- xml2rfc-linkmailto [5](#)
- xml2rfc-private [5](#)
- xml2rfc-sortrefs [5](#)
- xml2rfc-symrefs [5](#)
- xml2rfc-toc [5](#)
- xml2rfc-tocdepth [5](#)
- xml2rfc-topblock [5](#)
- parse-xml-in-artwork PI pseudo-attribute [6](#)
- private PI pseudo-attribute [5](#)
- Processing Instruction pseudo attributes
  - allow-markup-in-artwork [6](#)
  - authors-section [6](#)
  - background [5](#)
  - comments [5](#)
  - compact [5](#)
  - editing [5](#)
  - footer [5](#)
  - header [5](#)

- [include](#) [5](#)
- [inline](#) [5](#)
- [iprnotified](#) [5](#)
- [linkmailto](#) [5](#)
- [needLines](#) [5](#)
- [parse-xml-in-artwork](#) [6](#)
- [private](#) [5](#)
- [sec-no-trailing-dots](#) [6](#)
- [slides](#) [5](#)
- [sortrefs](#) [5](#)
- [strict](#) [5](#)
- [subcompact](#) [5](#)
- [support-rfc2731](#) [6](#)
- [symrefs](#) [5](#)
- [toc](#) [5](#)
- [tocdepth](#) [5](#)
- [tocindent](#) [5](#)
- [tocompact](#) [5](#)
- [topblock](#) [5](#)
- [xml2rfc-ext-parse-xml-in-artwork](#) [parameter](#) [6](#)
- [xml2rfc-ext-sec-no-trailing-dots](#) [parameter](#) [6](#)
- [xml2rfc-ext-support-rfc2731](#) [parameter](#) [6](#)
- [xml2rfc-footer](#) [parameter](#) [5](#)
- [xml2rfc-header](#) [parameter](#) [5](#)
- [xml2rfc-inline](#) [parameter](#) [5](#)
- [xml2rfc-iprnotified](#) [parameter](#) [5](#)
- [xml2rfc-linkmailto](#) [parameter](#) [5](#)
- [xml2rfc-private](#) [parameter](#) [5](#)
- [xml2rfc-sortrefs](#) [parameter](#) [5](#)
- [xml2rfc-symrefs](#) [parameter](#) [5](#)
- [xml2rfc-toc](#) [parameter](#) [5](#)
- [xml2rfc-tocdepth](#) [parameter](#) [5](#)
- [xml2rfc-topblock](#) [parameter](#) [5](#)
- [xsltproc](#) [8](#)

## R

- [Relation.Replaces](#) [DCMI property](#) [10](#)
- [rfc.abstract](#) [anchor](#) [7](#)
- [rfc.authors](#) [anchor](#) [7](#)
- [rfc.copyright](#) [anchor](#) [7](#)
- [rfc.copyrightnotice](#) [anchor](#) [7](#)
- [rfc.figure.n](#) [anchor](#) [7, 7](#)
- [rfc.figure.u.n](#) [anchor](#) [7, 7](#)
- [rfc.index](#) [anchor](#) [7](#)
- [rfc.ipr](#) [anchor](#) [7](#)
- [rfc.iref.n](#) [anchor](#) [7](#)
- [rfc.note.n](#) [anchor](#) [7](#)
- [rfc.references](#) [anchor](#) [7](#)
- [rfc.references.n](#) [anchor](#) [7](#)
- [rfc.section.n](#) [anchor](#) [7](#)
- [rfc.section.n.p.m](#) [anchor](#) [7](#)
- [rfc.status](#) [anchor](#) [7](#)
- [rfc.toc](#) [anchor](#) [7](#)

## S

- [Safari](#) [8](#)
- [Saxon](#) [8](#)
- [sec-no-trailing-dots](#) [PI pseudo-attribute](#) [6](#)
- [slides](#) [PI pseudo-attribute](#) [5](#)
- [sortrefs](#) [PI pseudo-attribute](#) [5](#)
- [strict](#) [PI pseudo-attribute](#) [5](#)
- [subcompact](#) [PI pseudo-attribute](#) [5](#)
- [support-rfc2731](#) [PI pseudo-attribute](#) [6](#)
- [symrefs](#) [PI pseudo-attribute](#) [5](#)

## T

- [toc](#) [PI pseudo-attribute](#) [5](#)
- [tocdepth](#) [PI pseudo-attribute](#) [5](#)
- [tocindent](#) [PI pseudo-attribute](#) [5](#)
- [tocompact](#) [PI pseudo-attribute](#) [5](#)
- [topblock](#) [PI pseudo-attribute](#) [5](#)

## X

- [Xalan](#) [8](#)
- [xml-stylesheet](#) [PI](#) [9](#)
- [xml2rfc-background](#) [parameter](#) [5](#)
- [xml2rfc-comments](#) [parameter](#) [5](#)
- [xml2rfc-editing](#) [parameter](#) [5, 5](#)
- [xml2rfc-ext-allow-markup-in-artwork](#) [parameter](#) [6](#)
- [xml2rfc-ext-authors-section](#) [parameter](#) [6](#)