
About DocBook

Prof. Jean-François Perrot

Table of Contents

1. Principle	2
1.1. What is DocBook ?	2
1.2. Main features	2
2. Stylesheets	3
2.1. Availability	3
2.2. Usage through customization	3
2.3. Use a CSS to easily customize HTML output	4
2.4. The basic technique is overriding imported rules	5

A first approach to DocBook and its use for project documentation.



The DocBook Schema Version

OASIS Standard

1 November 2009

1. Principle

1.1. What is DocBook ?

DocBook is an XML formalism for describing (technical) complex documents intended for publication.

This description is independent of the publication medium : web page (HTML), printed matter (PDF), epub or other.

Once a publication medium is chosen, the XML-DocBook document will be transformed into the corresponding XML format via an XSLT stylesheet.

1.1.1. Standards

The DocBook standard is managed by the OASIS consortium [<https://www.oasis-open.org>]. The current version is **version 5** [<http://docs.oasis-open.org/docbook/specs/docbook-5.0-spec-os.html>].

Since version 5,

- DocBook uses the namespace "<http://docbook.org/ns/docbook>"
- DocBook structure is officially specified by an RNG grammar, to be found here [<http://docs.oasis-open.org/docbook/rng/5.0/>].

Some explanations about this grammar are given in § 3.2. *DocBook Schema Structure* [<http://www.docbook.org/tdg5/en/html/ch05.html#ch05-patnames>] of *DocBook 5: The Definitive Guide*, chap. 5.

- There is also a "hand-crafted", non-normative XML Schema, to be found here [<http://www.docbook.org/xsd/5.1CR3/>].
“ DocBook XSD V5.1CR3 represents an attempt to hand-craft a useful XML Schema for DocBook V5.1. This schema remains non-normative, and more permissive than the normative DocBook schema, but (hopefully) represents a significant improvement over the previous, machine-generated attempt. [DocBook.org [<http://www.docbook.org>]] ”

1.1.2. References

DocBook 5: The Definitive Guide [<http://www.docbook.org>] : the source of all knowledge. **Online version** [<http://www.docbook.org/tdg5/en/html/docbook.html>] .

1.2. Main features

- *Like HTML*

DocBook normalizes spaces and ignores line breaks.
DocBook distinguishes *block* and *inline* elements.

- *Unlike HTML*

“DocBook does not provide the equivalent of HTML's br tag, so there's no way to interject a line break into normal running text.”

See `literallayout`.

DocBook adds many more *block* and *inline* elements.

DocBook provides two different *root* elements : `book` and `article`.

2. Stylesheets

2.1. Availability

A set of standard stylesheets is available from a SourceForge project called *DocBook XSL*.

Download (zipped directory) [<http://sourceforge.net/projects/docbook/files/docbook-xsl-ns/>]. Inside of the `docbook-xsl-ns-1.78.1` directory, there is a subdirectory for each publication medium, and in each of these subdirectories the master stylesheet is `docbook.xsl`.

Example : `docbook-xsl-ns-1.78.1/xhtml/docbook.xsl`.

Explanations by DocBook main designer : *The Design of the DocBook XSL Stylesheets* [<http://nwalsh.com/docs/articles/dbdesign/>].

2.2. Usage through customization

It is up to the user to **customize the stylesheet** that fits the publishing medium of his choice in order to suit his specific needs.

Reference documentation [<http://docbook.sourceforge.net/release/xsl/current/doc/>].

Users' guide [<http://www.sagehill.net/docbookxsl/index.html>].

The main tool for customizing a DocBook-XSL stylesheet (call it `stls.xsl`) is

1. to *import* `stls.xsl`;
2. to add xsl elements that will modify `stls.xsl`, namely:

- set xsl parameters,
- fill in some attributes,
- override some rules from `stls.xsl`.

Example for PDF output (`doc2fo.xsl`)

```
<?xml version='1.0'?>
<!DOCTYPE stylesheet [
  <!ENTITY dbdir "/path-to-reach.../docbook-xsl-ns-1.78.1">
]>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  version="1.0"
  xmlns:d="http://docbook.org/ns/docbook"
  xmlns:fo="http://www.w3.org/1999/XSL/Format">

  <xsl:import href= "&dbdir;/fo/docbook.xsl"/>

  <xsl:output method="xml" indent="yes"/>

  <!-- setting paper size -->
  <xsl:param name="paper.type" select="'A4'"/>

  <!-- titles with the same typeface as the text (default is 'sans-serif') -->
  <xsl:param name="title.font.family" select="'serif'"/>

  <!-- ensuring a page break before first-level sections -->
  <xsl:attribute-set name="section.level1.properties">
    <xsl:attribute name="break-before">page</xsl:attribute>
  </xsl:attribute-set>
```

```
<!-- special rule for quotations -->
<xsl:template match="d:quote">
  <fo:block font-family="sans-serif">
    <xsl:apply-imports />
  </fo:block>
</xsl:template>

</xsl:stylesheet>
```

2.3. Use a CSS to easily customize HTML output

The DocBook-to-XHTML stylesheet translates almost verbatim DocBook tag names into HTML-CSS class names.

Example : DocBook source, XHTML rendering

```
<simplelist>
  <member>
    <quote>DocBook does not provide the equivalent of HTML's br tag,
      so there's no way to interject a line break into normal running text.
    </quote>
    See <computeroutput>literallayout</computeroutput>.
  </member>
  <member>DocBook adds many more <emphasis>block</emphasis> and <emphasis>inl
  <member>DocBook provides two different <emphasis>root</emphasis> elements :
    <computeroutput>book</computeroutput> and <computeroutput>article</compute
  </member>
</simplelist>
```

```
-----

<table border="0" summary="Simple list" class="simplelist">
  <tr>
    <td>
      <td>
        <span class="quote"><span class="quote">DocBook does not provide the e
          so there's no way to interject a line break into normal running text.
        </span></span>
        See <code class="computeroutput">literallayout</code>.
      </td>
    </tr>
    <tr>
      <td>DocBook adds many more <span class="emphasis"><em>block</em></span>
    </tr>
    <tr>
      <td>DocBook provides two different <span class="emphasis"><em>root</em>
        <code class="computeroutput">book</code> and <code class="computeroutput
      </td>
    </tr>
  </table>
```

This opens the possibility to attach specific styles to DocBook tags via a CSS.

Attachment of a CSS is obtained by setting the `html.stylesheet` parameter to the URL of the CSS :

```
<xsl:param name="html.stylesheet" select="'myCSS.css'"/>
```

Very simple example :

```
body {
  margin-left:50px;
  font-family:sans-serif;
}
.quote {
  font-family:serif;
}
```

2.4. The basic technique is overriding imported rules

Use of CSS applies only to XHTML, not to PDF output.

Use of CSS applies only to style, not to inserting elements such as horizontal bar `<hr />`.

So that overriding of imported rules remains the major tool for the DocBook designer !

Example for HTML output (doc2xhtml.xsl)

```
<?xml version='1.0'?>
<!DOCTYPE stylesheet [
  <!ENTITY dbdir "/Users/jfp/Sites/EPITA/International/Site2014b/DocBook/docbook" />
]>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  version="1.0"
  xmlns:d="http://docbook.org/ns/docbook"
  xmlns="http://www.w3.org/1999/xhtml">

  <xsl:import href="&dbdir;/xhtml/docbook.xsl"/>

  <xsl:output method="xml" indent="yes"/>

  <!-- setting the CSS -->
  <xsl:param name="html.stylesheet" select="'myCSS.css'"/>

  <!-- inserting a bar before each sect1 -->
  <xsl:template match="d:sect1"> <!-- override -->
    <br />
    <hr />
    <xsl:apply-imports />
  </xsl:template>

</xsl:stylesheet>
```