
XMLmind XML Editor - DocBook Support

Hussein Shafie, Pixware

December 2, 2005

Table of Contents

1. DocBook menu	1
1.1. Convert Document sub-menu	2
2. Custom bindings	5
3. Table rendering	5
3.1. HTML tables	6

The DocBook specific commands described below are also used by the Simplified DocBook and Slides document types.

1. DocBook menu

Table editing commands fully support DocBook (CALS) tables.

Insert Column Before

Implicitly or explicitly selected element must be an `entry` or the descendant of an `entry`. Inserts a column before column containing specified cell.

A table cell is specified by implicitly or explicitly selecting the table cell or any of its descendants.

Insert Column After

Implicitly or explicitly selected element must be an `entry` or the descendant of an `entry`. Inserts a column after column containing specified cell.

Delete Column

Implicitly or explicitly selected element must be an `entry` or the descendant of an `entry`. Deletes column containing specified cell.

Insert Row Before

Implicitly or explicitly selected element must be an `entry` or the descendant of an `entry`. Inserts a row before row containing specified cell.

Rows editing commands are enabled, not only by implicitly or explicitly selecting a table cell or any of its descendants, but also by explicitly selecting a table row.

Insert Row After

Implicitly or explicitly selected element must be an `entry` or the descendant of an `entry`. Inserts a row before row containing specified cell.

Delete Row

Implicitly or explicitly selected element must be an `entry` or the descendant of an `entry`. Deletes row containing specified cell.

»§ Increase Section Level

Requires one or several `section`, `sect1`, `sect2`, `sect3`, `sect4` or `sect5` elements to be explicitly selected. This command will increase the level of selected sections by giving them a new parent section.

«§ Decrease Section Level

Requires one or several `section`, `sect1`, `sect2`, `sect3`, `sect4` or `sect5` elements to be explicitly selected. This command will decrease the level of selected sections by moving them out of their parent section.

🔗 Convert informal<element> to/from <element>

Converts an ``informal element" to/from a ``formal element" having a title.

This command currently works for `informaltable/table`, `informalfigure/figure` and `informalexample/example`.

Go to Opposite Link End

An element with an `ID`, `IDREF` or `IDREFS` attribute must be implicitly or explicitly selected. Scrolls to and selects opposite link end (if found). For example, will scroll to and select `<section id="foo">` if the command has been invoked with `<link linkend="foo">` selected.

Go to Next Link

An element with an `IDREF` or `IDREFS` attribute must be implicitly or explicitly selected. Scrolls to and selects next similar link (if found). For example, will scroll to and select following `<callout arearefs="foo bar">` if the command has been invoked with `<link linkend="foo">` selected.

Go to Previous Link

An element with an `IDREF` or `IDREFS` attribute must be implicitly or explicitly selected. Scrolls to and selects previous similar link (if found). For example, will scroll to and select preceding `<link linkend="foo">` if the command has been invoked with `<link linkend="foo">` selected.

1.1. Convert Document sub-menu

Convert to HTML..., Convert to HTML (one page)..., Convert to HTML (one page, no TOC)...

Converts the document being edited to multi page or single page HTML.

Convert to RTF..., Convert to RTF (no TOC)...

Converts the document being edited to RTF (Rich Text Format) using XMLmind FO Converter Personal Edition (see <http://www.xmlmind.com/foconverter>).

The RTF document generated by this command can be edited and printed using Microsoft® Word 2000 and above.

Convert to WordprocessingML..., Convert to WordprocessingML (no TOC)...

Converts the document being edited to WordprocessingML using XMLmind FO Converter Personal Edition (see <http://www.xmlmind.com/foconverter>).

The WordprocessingML document generated by this command can be edited and printed using Microsoft® Word 2003 and above.

Print PostScript..., Print PostScript (no TOC)...

Converts the document being edited to PostScript® using RenderX XEP (see <http://www.renderx.com/>), if its plug-in has been installed, and Apache FOP otherwise (see <http://xml.apache.org/fop>), and then, sends the generated file to the chosen printer.

Convert to PDF..., Convert to PDF (no TOC)...

Converts the document being edited to PDF (Adobe® Portable Document Format, also known as Acrobat®) using RenderX XEP (see <http://www.renderx.com/>), if its plug-in has been installed, and Apache FOP otherwise (see <http://xml.apache.org/fop>).

The last 8 commands, as any command using a FO processor plug-in, need XMLmind XML Editor Professional Edition in order to be executed successfully.

All the above **Convert** commands display the URL chooser dialog box rather than the standard file chooser dialog box.

For all **Convert** commands except for the "**Convert to HTML**" command, you must specify the URL (Uniform Resource Locator) of a save file. The "**Convert to HTML**" command creates multiple HTML pages with a first page called `index.html`, therefore you need to specify the URL of a save directory.

Note that these commands can create directories on the fly, if needed to. For example, if you specify `http://www.acme.com/docs/report43/mydoc.html` as the URL of the save file and if directory `report43/` does not exist, this directory will be created during command execution.

Uploading the files generated by these commands to a FTP or WebDAV server is only possible with XMLmind XML Editor Professional Edition.

Customizing the XSL style sheets used by the above Convert commands


The XSL style sheets used to convert DocBook, Simplified DocBook and Slides document to HTML, RTF, PostScript or PDF can be customized by specifying parameters, which are name/value pairs.

The parameters of the XSL style sheets used for DocBook and Simplified DocBook are documented in "DocBook XSL Stylesheet Documentation". The parameters of the XSL style sheets used for Slides are documented in "The Slides Document Type".

The easiest way to specify your own parameters is to add a named `parameterGroup` configuration element to your `customize.xxe` file:

1. Open in XXE `%SystemDrive%\Documents and Settings\%USERNAME%\Application Data\XMLmind\XMLEditor\addon\customize.xxe` (~/.xxe/addon/customize.xxe on Linux and Mac) if you have already created this file.



If this file does not exist, create it with XXE using **File|New, XMLmind XML Editor Configuration|Template**.

2. In the `cfg:configuration` bar, click on the  icon and choose `cfg:parameterGroup` from the menu.
3. Give its name to the `parameterGroup` by filling the **name** field, example: `docb.toPS.transformParameters`.

The names of the `parameterGroups` supported by the Convert commands is found in the following tables:

DocBook Convert command	Name of associated parameterGroup
Convert to HTML	<code>docb.toHTML.transformParameters</code>
Convert to HTML (one page)	<code>docb.toHTML1.transformParameters</code>
Convert to RTF	<code>docb.toRTF.transformParameters</code>
Print PostScript, Convert to PDF	<code>docb.toPS.transformParameters</code>

Slides Convert command	Name of associated parameterGroup
Convert to HTML	<code>slides.toHTML.transformParameters</code>
Convert to RTF	<code>slides.toRTF.transformParameters</code>
Print PostScript, Convert to PDF	<code>slides.toPS.transformParameters</code>

4. In the `cfg:parameterGroup` bar, click on the  icon and choose `cfg:parameter` from the menu.
 - a. Type the name of the parameter in the **name** field. Example: `shade.verbatim`.
 - b. Type the value of the parameter in the text placeholder below the **name** field. Example: 0.
5. If you need to add more `cfg:parameter` elements, in the last `cfg:parameter` element, click on the  icon, select `cfg:parameter` from the menu and so on.
6. Save `customize.xxe`, quit XXE and restart the editor to force it to reload your new/modified `customize.xxe`.
7. Test your customization by using the newly parametrized Convert command.

Note

It is also possible to *extensively* customize the **Convert** commands by specifying alternate XSLT style sheets for them. This is explained in the documentation of the `process/transform` configuration element found in XMLmind XML Editor - Commands.

2. Custom bindings

Keystroke	Action
Enter	Inside a <code>para/simpara</code> , splits the <code>para/simpara</code> in two parts. Elsewhere, normal behaviour.
Del	At the end of a <code>para/simpara</code> , if there is no node or text selection, joins this <code>para/simpara</code> with following sibling <code>para/simpara</code> if any. Otherwise, normal behaviour.
BackSpace	At the beginning of a <code>para/simpara</code> , if there is no node or text selection, joins this <code>para/simpara</code> with preceding sibling <code>para/simpara</code> if any. Otherwise, normal behaviour.
Ctrl-Enter	Inside a <code>para</code> , <code>simpara</code> , <code>listitem</code> , <code>callout</code> , <code>step</code> , inserts same element after this one. Elsewhere, no effect. Note If the <code>para/simpara</code> is the first child of a <code>listitem</code> , <code>callout</code> or <code>step</code> , it is a <code>listitem</code> , <code>callout</code> or <code>step</code> which is inserted, not a <code>para/simpara</code> . If, in such case, you want to insert a new <code>para/simpara</code> , simply press the Enter key at the end of the paragraph.
Shift-Ctrl-Enter	Inside a <code>para</code> , <code>simpara</code> , <code>listitem</code> , <code>callout</code> , <code>step</code> , inserts same element before this one. Elsewhere, no effect.
Application Event	Action
file-drop	On an <code>ulink</code> element, changes the value of attribute <code>url</code> to the URL of the dropped file. Elsewhere, opens the dropped file in the editor.

3. Table rendering

The following attributes are either completely ignored or partially supported. All other attributes are supported.

Attribute	Support
<code>table</code> (or <code>informaltable</code>) <code>orient</code>	Ignored.

Attribute	Support
table (or informaltable) pgwide	Ignored.
colspec colwidth	All forms including "2*" or "3*+1pc" are supported. Coefficients of "*" are always converted to integers. Examples: "2.5*" is equivalent to "2*". "3.95*+0.5in" is equivalent to "4*+0.5in". A column must contain at least one cell with a column span equal to 1 for the colwidth attribute to have an effect.
entry rotate	Ignored.
align	Values justify and char are rendered like left.
char	Ignored. See align.
charoff	Ignored. See align.

3.1. HTML tables

DocBook supports HTML tables as well as CALS tables (that is, ``traditional" DocBook tables) starting from version 4.3. Therefore XMLmind XML Editor also supports both table models. See XMLmind XML Editor - XHTML Support for details.

The only limitation is that mixing both HTML and CALS content models in the same `table` or `informaltable` is *absolutely not supported* by table rendering code and by table editing commands, even if this is allowed according to the DTD V4.3.

Example 1: an `informaltable` contains `tr` child elements. In such case, the `informaltable` is an HTML table. Setting attribute `frame` to `topbot` on this `informaltable` will have absolutely no visual effect.

Example 2: a `table` has a child `tgroup` element which itself contains a `tbody` with `row/entry` descendants. In such case, the `table` is a CALS table. Adding a `thead` having `tr/td` descendants before the `tbody` of the `tgroup` would lead to catastrophic results. Fortunately, the DocBook configuration of XMLmind XML Editor makes it hard to do this unintentionally.