

Release Notes

DocBook Project XSL Stylesheets

COLLABORATORS

	<i>TITLE :</i> Release Notes		<i>REFERENCE :</i>
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		May 17, 2006	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Release: 1.70.0	5
1.1	Common	5
1.2	Extensions	6
1.3	FO	6
1.4	HTML	10
1.5	Manpages	13
1.6	Params	17
1.7	Profiling	21
1.8	Tools	22
1.9	WordML	22
2	Release 1.69.1	22
3	Release 1.69.0	22
3.1	Common	22
3.2	FO	23
3.3	Help	24
3.4	HTML	24
3.5	man	24
4	Release 1.68.1	29
5	Release 1.68.0	29
6	Release 1.67.2	30
7	Release 1.67.1	31
8	Release 1.67.0	32
9	Release 1.66.1	33
10	Release 1.65.0	35
11	Release 1.64.1	35
12	Release 1.61.0	35
13	Release 1.60.1	36
14	Release 1.59.2	38
15	Release 1.59.1	38

16	Release 1.58.0	38
17	Release 1.57.0	39
18	Release 1.56.0	39
19	Older releases	40
20	About dot-zero releases	40

These are the release notes for the DocBook XSL Stylesheets distribution. This document lists enhancements and changes to the public APIs (user-configurable parameters) and generally excludes descriptions of bug fixes, which are instead documented in a separate [NEWS](#) file.

1 Release: 1.70.0

As with all DocBook Project [dot-zero](#) releases, this is an experimental release. It will be followed shortly by a stable release.

This release adds a number of new features, including:

- support for selecting alternative index-collation methods (in particular, support for using a collation library developed by Eliot Kimber)
- improved handling of DocBook 5 document instances (through a namespace-stripping mechanism)
- full support for CALS and HTML tables in manpages output
- a mechanism for preserving relative URIs in documents that make use of XInclude
- support for the "new" .90 version of FOP
- enhanced capabilities for controlling formatting of lists in HTML and FO output
- autogeneration of AUTHOR and COPYRIGHT sections in manpages output
- support for generating crop marks in FO/PDF output

The following is a detailed list of enhancements and API changes that have been made since the 1.69.1 release.

1.1 Common

The following changes have been made to the `common` code since the 1.69.1 release.

- Add the `xsl:key` for the kimber indexing method.
- Add support for `qandaset`.
- Support `dbhtml/dbfo` start PI for `orderedlist` numbering in both HTML and FO
- Added CVS header.
- Changed content model of text element to ANY rather than #PCDATA because they could contain markup.
- Added `refentry.meta.get.quietly` param.

If zero (the default), notes and warnings about "missing" markup are generated during gathering of `refentry` metadata. If non-zero, the metadata is gathered "quietly" -- that is, the notes and warnings are suppressed.

NOTE: If you are processing a large amount of `refentry` content, you may be able to speed up processing significantly by setting a non-zero value for `refentry.meta.get.quietly`.

- After namespace stripping, the source document is the temporary tree created by the stripping process and it has the wrong base URI for relative references. Earlier versions of this code used to try to fix that by patching the elements with relative `@fileref` attributes. That was inadequate because it calculated an absolute base URI without considering that there might be `xml:base` attributes already in effect. It seems obvious now that the right thing to do is simply to put the `xml:base` on the root of the document. And that seems to work.
- Added support for "software" and "sectdesc" class values on `refmiscinfo`; "software" is treated identically to "source", and "setdesc" is treated identically to "manual".
- Added support for DocBook 5 namespace-stripping in manpages stylesheet. Closes request #1210692.
- Added `<xsl:template match="/">` to make `stripns.xsl` usable as a standalone stylesheet for stripping out DocBook 5/NG to DocBook 4. Note that DocBook XSLT drivers that include this stylesheet all override the `match="/"` template.
- Number figures, examples, and tables from `book` if there is no prefix (i.e. if `chapter.autolabel` is set to 0). This avoids having the list of figures where the figures mysteriously restart their numeration periodically when `chapter.autolabel` is set to 0.
- Add `task` template in `title.markup` mode.
- Add children (with ids) of formal objects to target data.
- Added support for case when `personname` doesn't contain specific name markup (as allowed in DocBook 5.0)

1.2 Extensions

The following changes have been made to the `extensions` code since the 1.69.1 release.

- Support Xalan 2.7
- Handle the case where the `imageFn` is actually a URI. This still needs work.

1.3 FO

The following changes have been made to the `fo` code since the 1.69.1 release.

- Adapted to the new indexing code. Now works just like a wrapper that calls `kosek` indexing method, originally implemented here.
- Added parameters for header/footer `table` minimum height.
- Add the `index.method` parameter.
- Integrate support for three indexing methods:
 - the original English-only method.
 - Jirka Kosek's method using EXSLT extensions.
 - Eliot Kimber's method using Saxon extensions.Use the `'index.method'` parameter to select.
- Add support for TOC for `qandaset` in `fo` output.
- Added parameter `ulink.hyphenate.chars`.
Added parameter `insert.link.page.number`.

- Implemented feature request #942524 to add `insert.link.page.number` to allow link element cross references to have a page number.
- Add support for `ulink.hyphenate.chars` so more characters can be break points in urls.
- Implemented patch #1075144 to make the url text in a ulink in FO output an active link as well.
- `table` footnotes now have their own `table.footnote.properties` attribute set.
- Add `qandaset` to `root.elements`.
- Added `mode="page.sequence"` to make it easier to put content into a page sequence. First used for `qandaset`.
- Implemented feature request #1434408 to support formatting of `biblioentry`.
- Added `biblioentry.properties`.
- Support PTC/Arbortext bookmarks
- Added `table.footnote.properties` to permit `table` footnotes to format differently from regular footnotes.
- Refactored `table` templates to unify their processing and support all options in all types. Now `table` and `informaltable`, in both Cals and Html markup, use the same templates where possible, and all support `pgwide`, rotation, and floats. There is also a placeholder `table.container` template to support wrapping a `table` in a layout `table`, so the XEP `table` title "continued" extension can be more easily implemented.
- Added new attribute set `toc.line.properties` for controlling appearance of lines in ToC/LoT
- Added support for float to `example` and `equation`. Added support for `pgwide` to `figure`, `example`, and `equation` (the latter two via a `dbfo pgwide="1"` processing instruction).
- Add `pgwide.properties` attribute-set.
- Added `refclass.suppress` param.

If the value of `refclass.suppress` is non-zero, then display `refclass` contents is suppressed in output. Affects HTML and FO output only.

- Improved support for `task` subelements
- Adjusted spacing around K&R-formatted `Funcdef` and `Paramdef` output such that it can more easily be discerned where one ends and the other begins. Closes #1213264.
- Made handling of `paramdef/parameter` in FO output consistent with that in HTML and manpages output. Closes #1213259.
- Made handling of `Refnamediv` consistent with formatting in HTML and manpages output; specifically, changed so that `Refname` (comma-separated list of multiple instances found) is used (instead of `Refentrytitle` as previously), then em-dash, then the `Refpurpose`. Closes #1212562.

- Added output of `Releaseinfo` to recto titlepage ("copyright" page) for `Book` in FO output. This makes it consistent with HTML output. Closes #1327034. Thanks to Paul DuBois for reporting.
- Added condition for setting `block-progression-dimension.minimum` on table-row, instead of height, when `fop1.extensions` is non-zero. For an explanation of the reason for the change, see: <http://wiki.apache.org/xmlgraphics-fop/Troubleshooting/CommonLogMessages>
- Added new `refclass.suppress` param for suppressing display of `Refclass` in HTML and FO output. Did not add it to manpages because manpages stylesheet is currently just silently ignoring `Refclass` anyway. Closes request #1461065. Thanks to Davor Ocelic (docelic) for reporting.
- Add support for keep-together PI to informal objects.
- Add support for `fop1.extensions`.
- Add support for `fop1` bookmarks.
- Add `fop1.extensions` parameter to add support for fop development version.
- Start supporting fop development version, which will become fop version 1.
- Add template for `task` in `mode="xref-to"`.
- `table` footnotes now also get `footnote.properties` attribute-set.
- Added `index.separator` named template to compute the separator punctuation based on locale.
- Added support for `link`, `olink`, and `xref` within `OO Classsynopsis` and children. (Because DocBook NG/5 allows it).
- Support `date` as an inline
- Added new parameter `keep.relative.image.uris`
- Map Unicode space characters U+2000-U+200A to fo:leaders.
- Output a real em dash for em-dash dingbat (instead of two hypens).
- Support default `label` width parameters for itemized and ordered lists
- Generate localized `title` for `Refsynopsisdiv` if no appropriate `Title` descendant found in source. Closes #1212398. This change makes behavior for the `Synopsis title` consistent with the behavior of HTML and manpages output.

Also, added `xsl:use-attribute-sets="normal.para.spacing"` to block generated for `Cmdsynopsis` output. Previously, that block had no spacing at all specified, which resulted it being crammed up to closely to the `Synopsis` head.
- Added parameters to support localization of `index` item punctuation.
- Added `index.number.separator`, `index.range.separator`, and `index.term.separator` parameters to support localization of punctuation in `index` entries.

- Added "Cross References" section in HTML doc (for consistency with the FO doc). Also, moved the existing FO "Cross References" section to follow the "Linking" section.
- Added ID attributes to all `Reference` elements (e.g., `id="tables"` for the doc for section on Table params). So pages for all subsections of ref docs now have stable filenames instead of arbitrary generated filenames.
- Added two new parameters for handling of multi-term `varlistentry` elements:

variablelist.term.break.after:

When the *variablelist.term.break.after* is non-zero, it will generate a line break after each term multi-term `varlistentry`.

variablelist.term.separator:

When a `varlistentry` contains multiple term elements, the string specified in the value of the *variablelist.term.separator* parameter is placed after each term except the last. The default is ", " (a comma followed by a space). To suppress rendering of the separator, set the value of *variablelist.term.separator* to the empty string ("").

These parameters are primarily intended to be useful if you have multi-term `varlistentries` that have long terms.

Closes #1306676. Thanks to Sam Steingold for providing an example "lots of long terms" doc that demonstrated the value of having these options.

Also, added `normalize-space()` call to processing of each term.

This change affects all output formats (HTML, PDF, manpages). The default behavior should pretty much remain the same as before, but it is possible (as always) that the change may introduce some new bugginess.

- Add sidebar titlepage placeholder attset for styles.
- Add titlepage for sidebar.
- Implemented RFE #1292615.

Added bunch of new parameters (attribute sets) that affect list presentation: `list.block.properties`, `itemizedlist.properties`, `orderedlist.properties`, `itemizedlist.label.properties` and `orderedlist.label.properties`. Default behaviour of stylesheets has not been changed but further customizations will be much more easier.

- Implemented RFE #1242092.

You can enable crop marks in your document by setting `crop.marks=1` and `xep.extensions=1`.

Appearance of crop marks can be controlled by parameters *crop.mark.bleed* (6pt), *crop.mark.offset* (24pt) and *crop.mark.width* (0.5pt).

Also there is new named template called `user-xep-pis`. You can overwrite it in order to produce some PIs that can control XEP as described in http://www.renderx.com/reference.html#Output_Formats

1.4 HTML

The following changes have been made to the `html` code since the 1.69.1 release.

- implemented `index.method` parameter and three methods.
- added `index.method` parameter to support 3 indexing methods.
- Implemented feature request #1072510 as a processing instruction to permit including external HTML content into HTML output.
- Added new parameter `chunk.tocs.and.lots.has.title` which controls presence of `title` in a separate chunk with ToC/LoT. Disabling `title` can be very useful if you are generating frameset output (well, yes those frames, but some customers really want them ;-).
- Support `dbhtml/dbfo` start PI for `orderedlist` numbering in both HTML and FO
- Allow ToC without `title` also for `set` and `book`.
- Implemented floats uniformly for `figure`, `example`, `equation` and `informalfigure`, `informalexample`, and `informalequation`.
- Added the `autotoc.label.in.hyperlink` param.

If the value of `autotoc.label.in.hyperlink` is non-zero, labels are included in hyperlinked titles in the TOC. If it is instead zero, labels are still displayed prior to the hyperlinked titles, but are not hyperlinked along with the titles.

Closes patch #1065868. Thanks to anatoly techtonik for the patch.

- Added two new params: `html.head.legalnotice.link.types` and `html.head.legalnotice.link.multiple`.

If the value of the `generate.legalnotice.link` is non-zero, then the stylesheet generates (in the head section of the HTML source) either a single HTML link element or, if the value of the `html.head.legalnotice.link.multiple` is non-zero, one link element for each link type specified. Each link has the following attributes:

- a `rel` attribute whose value is derived from the value of `html.head.legalnotice.link.types`
- an `href` attribute whose value is set to the URL of the file containing the `legalnotice`
- a `title` attribute whose value is set to the `title` of the corresponding `legalnotice` (or a `title` programatically determined by the stylesheet)

For example:

```
<link rel="copyright" href="ln-id2524073.html" title="Legal Notice">
```

Closes #1476450. Thanks to Sam Steingold.

- Added `refclass.suppress` param.

If the value of `refclass.suppress` is non-zero, then display

`refclass` contents is suppressed in output. Affects HTML and FO output only.

- Improved support for `task` subelements
- Added new `refclass.suppress` param for suppressing display of `Refclass` in HTML and FO output. Did not add it to `manpages` because `manpages` stylesheet is currently just silently ignoring `Refclass` anyway. Closes request #1461065. Thanks to Davor Ocelic (docelic) for reporting.
- Process `alt` text with `normalize-space()`.
Replace tab indents with spaces.
- Content of `citation` element is automatically linked to the bibliographic entry with the corresponding `abbrev`.
- Add template for `task` in `mode="xref-to"`.
- Suppress ID warnings if the `.warnings` parameter is 0
- Add support for `floatstyle` to `figure`.
- Handling of `xref` to `area/areaset` need support in extensions code also. I currently have no time to touch extensions code, so code is here to be enabled when extension is fixed also.
- Added 3 parameters for overriding `gentext` for `index` punctuation.
- Added parameters to support localization of `index` item punctuation.
Added `index.separator` named template to compute the separator punctuation based on locale.
- Added a `<div class="{ $class }-contents">` wrapper around output of contents of all formal objects. Also, added an optional `<br class="{ class }-break"/>` linebreak after all formal objects.

WARNING: Because this change places an additional DIV between the DIV wrapper for the `equation` and the `equation` contents, it may break some existing CSS stylesheets that have been created with the assumption that there would never be an intervening DIV there.

The following is an example of what `Equation` output looks like as a result of the changes described above.

```
<div class="equation">
<a name="three" id="three"></a>

<p class="title"><b>(1.3)</b></p>

<div class="equation-contents">
<span class="mathphrase">1+1=3</span>
</div>
</div><br class="equation-break">
```

Rationale: These changes allow CSS control of the placement of the formal-object `title` relative to the formal-object contents. For example, using the CSS "float" property enables the `title` and contents to be rendered on the same line. Example stylesheet:

```
.equation {
```

```
margin-top: 20px;
margin-bottom: 20px;
}
.equation-contents {
float: left;
}
```

```
.equation .title {
margin-top: 0;
float: right;
margin-right: 200px;
}
```

```
.equation .title b {
font-weight: normal;
}
```

```
.equation-break {
clear: both;
}
```

Note that the purpose of the ".equation-break" class is to provide a way to clear off the floats.

If you want to instead have the `equation title` rendered to the left of the equation contents, you can do something like this:

```
.equation {
margin-top: 20px;
width: 300px;
margin-bottom: 20px;
}
.equation-contents {
float: right;
}
```

```
.equation .title {
margin-top: 0;
float: left;
margin-right: 200px;
}
```

```
.equation .title b {
font-weight: normal;
}
```

```
.equation-break {
clear: both;
}
```

- Added a `chunker.output.quiet` top-level parameter so that the chunker can be made quiet by default
- Added support for `link`, `olink`, and `xref` within `OO Classsynopsis` and children. (Because DocBook NG/5 allows it).
- New parameter: `id.warnings`. If non-zero, warnings are generated for titled objects that don't have titles. True by default; I wonder if this will be too aggressive?

- If the *keep.relative.image.uris* parameter is true, don't use the absolute URI (as calculated from *xml:base*) in the *img src* attribute, use the value the author specified. Note that we still have to calculate the absolute *filename* for use in the image intrinsics extension.
- Support date as an inline
- Added new parameter *keep.relative.image.uris*
- Added two new parameters for handling of multi-term *varlistentry* elements:

variablelist.term.break.after:

When the *variablelist.term.break.after* is non-zero, it will generate a line break after each *term* multi-term *varlistentry*.

variablelist.term.separator:

When a *varlistentry* contains multiple *term* elements, the string specified in the value of the *variablelist.term.separator* parameter is placed after each *term* except the last. The default is ", " (a comma followed by a space). To suppress rendering of the separator, set the value of *variablelist.term.separator* to the empty string ("").

These parameters are primarily intended to be useful if you have multi-term *varlistentries* that have long terms.

Closes #1306676. Thanks to Sam Steingold for providing an example "lots of long terms" doc that demonstrated the value of having these options.

Also, added *normalize-space()* call to processing of each *term*.

This change affects all output formats (HTML, PDF, manpages). The default behavior should pretty much remain the same as before, but it is possible (as always) that the change may introduce some new bugginess.

- Added "wrapper-name" param to *inline.charseq* named template, enabling it to output inlines other than just "span". Acronym and Abbrev templates now use *inline.charseq* to output HTML "acronym" and "abbr" elements (instead of "span"). Closes #1305468. Thanks to Sam Steingold for suggesting the change.

1.5 Manpages

The following changes have been made to the *manpages* code since the 1.69.1 release.

- Added the following params:

- *man.indent.width* (string-valued)
- *man.indent.refsect* (boolean)
- *man.indent.blurbs* (boolean)
- *man.indent.lists* (boolean)
- *man.indent.verbatims* (boolean)

Note that in earlier snapshots, *man.indent.width* was named *man.indentation.default.value* and the boolean params had names

like `man.indentation.*.adjust`. Also the `man.indent.blurbs` param was called `man.indentation.authors.adjust` (or something).

The behavior now is: If the value of a particular `man.indent.*` boolean param is non-zero, the corresponding contents (`refsect*`, list items, `authorblurb/personblurb`, `vervatims`) are displayed with a left margin indented by a width equal to the value of `man.indent.width`.

- Added `man.table.footnotes.divider` param.

In each `table` that contains footnotes, the string specified by the `man.table.footnotes.divider` parameter is output before the list of footnotes for the `table`.

- Added the `man.output.in.separate.dir`, `man.output.base.dir`, and `man.output.subdirs.enabled` parameters.

The `man.output.base.dir` parameter specifies the base directory into which man-page files are output. The `man.output.subdirs.enabled` parameter controls whether the files are output in subdirectories within the base directory.

The values of the `man.output.base.dir` and `man.output.subdirs.enabled` parameters are used only if the value of `man.output.in.separate.dir` parameter is non-zero. If the value of `man.output.in.separate.dir` is zero, man-page files are not output in a separate directory.

- Added `man.font.table.headings` and `man.font.table.title` params, for controlling font in `table` headings and titles.
- Added `man.font.funcsynopsisinfo` and `man.font.funcprototype` params, for specifying the roff font (for example, `BI`, `B`, `I`) for `funcsynopsisinfo` and `funcprototype` output.
- Added `man.segtitle.suppress` param.

If the value of `man.segtitle.suppress` is non-zero, then display of `segtitle` contents is suppressed in output.

- Added `man.output.manifest.enabled` and `man.output.manifest.filename` params.

If `man.output.manifest.enabled` is non-zero, a list of filenames for man pages generated by the stylesheet transformation is written to the file named by `man.output.manifest.filename`

- Added `refentry.meta.get.quietly` param.

If zero (the default), notes and warnings about "missing" markup are generated during gathering of `refentry` metadata. If non-zero, the metadata is gathered "quietly" -- that is, the notes and warnings are suppressed.

NOTE: If you are processing a large amount of `refentry` content, you may be able to speed up processing significantly by setting a non-zero value for `refentry.meta.get.quietly`.

- Changed names of all boolean indentation params to `man.indent.*`. Also discarded individual `man.indent.*.value` params and switched to just using a common `man.indent.width` param (3n by default).

- Added boolean `man.output.in.separate.dir` param, to control whether or not man files are output in separate directory.
- Added options for controlling indentation of verbatim output. Controlled through the `man.indentation.verbatims.adjust` and `man.indentation.verbatims.value` params. Closes #1242997
- Added options for controlling indentation in lists and in `*blurb` output in the `AUTHORS` section. Controlled through the `man.indentation.lists.adjust`, `man.indentation.lists.value`, `man.indentation.authors.adjust`, and `man.indentation.authors.value` parameters. Default is 3 characters (instead of the roff default of 8 characters). Closes #1449369.

Also, removed the indent that was being set on `informalexample` output. I will instead add an option for indenting verbatims, which I think is what the `informalexample` indent was intended for originally.

- Changed line-spacing call before `synopfragment` to use `".sp -1n"` ("n" units specified) instead of plain `".sp -1"`
- Added support for writing man files into a specific output directory and into appropriate subdirectories within that output directory. Controlled through the `man.base.dir` parameter (similar to the `base.dir` support in the HTML stylesheet) and the `man.subdirs.enabled` parameter, which automatically determines the name of an appropriate subdir (for example, `man/man7`, `man/man1`, etc.) based on the `section number/manvolnum` of the source `Refentry`.

Closes #1255036 and #1170317. Thanks to Denis Bradford for the original feature request, and to Costin Stroe for submitting a patch that was very helpful in implementing the support.

- Refined XPath statements and notification messages for `refentry` metadata handling.
- Added support for `copyright` and `legalnotice`. The manpages stylesheets now output a `COPYRIGHT` section, after the `AUTHORS` section, if a `copyright` or `legalnotice` is found in the source. The section contains the `copyright` contents followed by the `legalnotice` contents. Closes #1450209.
- Drastically reworked all of the XPath expressions used in `refentry` metadata gathering -- completely removed `$parentinfo` and turned `$info` into a set of nodes that includes the `*info` contents of the `Refentry` plus the `*info` contents all all of its ancestor elements. The basic XPath expression now used throughout is (using the example of checking for a `date`):

```
((($info[//date]))[last()]/date)[1].
```

That selects the "last" `*info/date` date in document order -- that is, the one either on the `Refentry` itself or on the closest ancestor to the `Refentry`.

It's likely this change may break some things; may need to pick up some pieces later.

Also, changed the default value for the `man.th.extra2.max.length` from 40 to 30.

- Added support for DocBook 5 namespace-stripping in manpages stylesheet. Closes request #1210692.
- Fixed handling of `table` footnotes.
With this checkin, the `table` support in the manpages stylesheet is now basically feature complete. So this change closes request #619532, "No support for tables" -- the oldest currently open manpages feature request, submitted by Ben Secrest (blsecres) on 2002-10-07. Congratulations to me [patting myself on the back].
- Added handling for `table` titles. Also fixed handling of nested tables; nest tables are now "extracted" and displayed just after their parent tables.
- Added option for turning off bold formatting in `Funcsynopsis`.
Boldface formatting in `function synopsis` is mandated in the `man(7)` man page and is used almost universally in existing man pages. Despite that, it really does look like crap to have an entire `Funcsynopsis` output in bold, so I added params for turning off the bold formatting and/or replacing it with a different roff special font (e.g., "RI" for alternating roman/italic instead of the default "BI" for alternating bold/italic). The new params are "`man.funcprototype.font`" and "`man.funcsynopsisinfo.font`". To be documented later.

Closes #1452247. Thanks to Joe Orton for the feature request.

- Use `AUTHORS` instead of `AUTHOR` if we have multiple people to attribute. Also, fixed checking such that we generate `author` section even if we don't have an `author` (as long as there is at least one other person/entity we can put in the section). Also adjusted assembly of content for `Author` metainfo field such that we now not only use `author`, but try to find a "best match" if we can't find an `author` name to put there.

Closes #1233592. Thanks to Sam Steingold for the request.

- Changes for request #1243027, "Improve handling of `AUTHOR` section."
This adds support for `Collab`, `Corpauthor`, `Corpcredit`, `Orgname`, `Publishername`, and `Publisher`. Also adds support for output of `Affiliation` and its children, and support for using `gentext` strings for auto-attributing roles (`Author`, `Editor`, `Publisher`, `Translator`, etc.). Also did a lot of code cleanup and modularization of all the `AUTHOR` handling code. And fixed a bug that was causing `Author` info to not be picked up correctly for metainfo comment we embed in man-page source.
- Support bold output for "`emphasis remap='B'`". (because Eric Raymond's `doclifter(1)` tool converts groff source marked up with ".B" request or "\fB" escapes to DocBook "`emphasis remap='B'`".)
- Added support for `Segmentedlist`.
Details: Output is tabular, with no option for "list" type output. Output for `Segtitle` elements can be suppressed by setting `man.segtitle.suppress`. If `Segtitle` content is output, it is rendered in italic type (not bold because not all terminals support bold and so italic ensures the stand out on those

terminals). Extra space (.sp line) at end of `table` code ensures that it gets handled correctly in the case where its source is the child of a `Para`. Closes feature-request #1400097. Thanks to Daniel Leidert for the patch and push, and to Alastair Rankine for filing the original feature request.

- Improved handling of `Author/Editor/Othercredit`.

Reworked content of (non-visible) comment added at top of each page (metadata stuff).

Added support for generating a *manifest* file (useful for cleaning up after builds, etc.)

- Added two new parameters for handling of multi-term `varlistentry` elements:

variablelist.term.break.after:

When the *variablelist.term.break.after* is non-zero, it will generate a line break after each `term` multi-term `varlistentry`.

variablelist.term.separator:

When a `varlistentry` contains multiple `term` elements, the string specified in the value of the *variablelist.term.separator* parameter is placed after each `term` except the last. The default is ", " (a comma followed by a space). To suppress rendering of the separator, set the value of *variablelist.term.separator* to the empty string ("").

These parameters are primarily intended to be useful if you have multi-term `varlistentries` that have long terms.

Closes #1306676. Thanks to Sam Steingold for providing an example "lots of long terms" doc that demonstrated the value of having these options.

Also, added `normalize-space()` call to processing of each `term`.

This change affects all output formats (HTML, PDF, manpages). The default behavior should pretty much remain the same as before, but it is possible (as always) that the change may introduce some new bugginess.

1.6 Params

The following changes have been made to the `params` code since the 1.69.1 release.

- New parameters to set header/footer `table` minimum height.
- Support multiple indexing methods for different languages.
- Remove `qandaset` and `qandadiv` from *generate.toc* for fo output because formerly it wasn't working, but now it is and the default behavior should stay the same.
- add support for page number references to `link` element too.
- Add support for more characters to hyphen on when *ulink.hyphenate* is turned on.

- New attribute-set to format `biblioentry` and `bibliomixed`.
- Added new parameter `chunk.tocs.and.lots.has.title` which controls presence of `title` in a separate chunk with ToC/LoT. Disabling `title` can be very useful if you are generating frameset output (well, yes those frames, but some customers really want them ;-).
- Added new attribute set `toc.line.properties` for controlling appearance of lines in ToC/LoT
- Allow `table` footnotes to have different properties from regular footnotes.
- Set properties for `pgwide="1"` objects.
- Added the `autotoc.label.in.hyperlink` param.

If the value of `autotoc.label.in.hyperlink` is non-zero, labels are included in hyperlinked titles in the TOC. If it is instead zero, labels are still displayed prior to the hyperlinked titles, but are not hyperlinked along with the titles.

Closes patch #1065868. Thanks to anatoly techtonik for the patch.

- Added two new params: `html.head.legalnotice.link.types` and `html.head.legalnotice.link.multiple`.

If the value of the `generate.legalnotice.link` is non-zero, then the stylesheet generates (in the head section of the HTML source) either a single HTML link element or, if the value of the `html.head.legalnotice.link.multiple` is non-zero, one link element for each link type specified. Each link has the following attributes:

- a `rel` attribute whose value is derived from the value of `html.head.legalnotice.link.types`

- an `href` attribute whose value is set to the URL of the file containing the `legalnotice`

- a `title` attribute whose value is set to the `title` of the corresponding `legalnotice` (or a `title` programatically determined by the stylesheet)

For example:

```
<link rel="copyright" href="ln-id2524073.html" title="Legal Notice">
```

Closes #1476450. Thanks to Sam Steingold.

- Added the following params:

- `man.indent.width` (string-valued)
- `man.indent.refsect` (boolean)
- `man.indent.blurbs` (boolean)
- `man.indent.lists` (boolean)
- `man.indent.verbatims` (boolean)

Note that in earlier snapshots, `man.indent.width` was named `man.indentation.default.value` and the boolean params had names like `man.indentation.*.adjust`. Also the `man.indent.blurbs` param was called `man.indentation.authors.adjust` (or something).

The behavior now is: If the value of a particular `man.indent.*` boolean param is non-zero, the corresponding contents (`refsect*`, list items, `authorblurb/personblurb`, `vervatims`) are displayed with a left margin indented by a width equal to the value of `man.indent.width`.

- Added `man.table.footnotes.divider` param.

In each `table` that contains footnotes, the string specified by the `man.table.footnotes.divider` parameter is output before the list of footnotes for the `table`.

- Added the `man.output.in.separate.dir`, `man.output.base.dir`, and `man.output.subdirs.enabled` parameters.

The `man.output.base.dir` parameter specifies the base directory into which man-page files are output. The `man.output.subdirs.enabled` parameter controls whether the files are output in subdirectories within the base directory.

The values of the `man.output.base.dir` and `man.output.subdirs.enabled` parameters are used only if the value of `man.output.in.separate.dir` parameter is non-zero. If the value of `man.output.in.separate.dir` is zero, man-page files are not output in a separate directory.

- Added `man.font.table.headings` and `man.font.table.title` params, for controlling font in `table` headings and titles.
- Added `man.font.funcsynopsisinfo` and `man.font.funcprototype` params, for specifying the roff font (for example, BI, B, I) for `funcsynopsisinfo` and `funcprototype` output.
- Changed to `select="0"` in `refclass.suppress` (instead of `..>0</..`)
- Added `man.segtitle.suppress` param.

If the value of `man.segtitle.suppress` is non-zero, then display of `segtitle` contents is suppressed in output.

- Added `man.output.manifest.enabled` and `man.output.manifest.filename` params.

If `man.output.manifest.enabled` is non-zero, a list of filenames for man pages generated by the stylesheet transformation is written to the file named by `man.output.manifest.filename`

- Added `refclass.suppress` param.

If the value of `refclass.suppress` is non-zero, then display `refclass` contents is suppressed in output. Affects HTML and FO output only.

- Added `refentry.meta.get.quietly` param.

If zero (the default), notes and warnings about "missing" markup are generated during gathering of `refentry` metadata. If non-zero, the metadata is gathered "quietly" -- that is, the notes and warnings are suppressed.

NOTE: If you are processing a large amount of `refentry` content, you may be able to speed up processing significantly by setting a non-zero value for `refentry.meta.get.quietly`.

- Added support for "software" and "sectdesc" class values on `refmiscinfo`; "software" is treated identically to "source", and "sectdesc" is treated identically to "manual".
- Drastically reworked all of the XPath expressions used in `refentry` metadata gathering -- completely removed `$parentinfo` and turned `$info` into a set of nodes that includes the `*info` contents of the `Refentry` plus the `*info` contents all all of its ancestor elements. The basic XPath expression now used throughout is (using the example of checking for a date):

```
(( $info[//date])[last()]/date)[1].
```

That selects the "last" `*info/date` date in document order -- that is, the one either on the `Refentry` itself or on the closest ancestor to the `Refentry`.

It's likely this change may break some things; may need to pick up some pieces later.

Also, changed the default value for the `man.th.extra2.max.length` from 40 to 30.

- Added option for turning off bold formatting in `Funcsynopsis`. Boldface formatting in `function synopsis` is mandated in the `man(7)` man page and is used almost universally in existing man pages. Despite that, it really does look like crap to have an entire `Funcsynopsis` output in bold, so I added params for turning off the bold formatting and/or replacing it with a different roff special font (e.g., "RI" for alternating roman/italic instead of the default "BI" for alternating bold/italic). The new params are "man.funcprototype.font" and "man.funcsynopsisinfo.font". To be documented later.

Closes #1452247. Thanks to Joe Orton for the feature request.

- `fop.extensions` now only for FOP version 0.20.5 and earlier.
- Support for `fop1` different from `fop` 0.20.5 and earlier.
- Reset default value to empty string so template uses `gentext` first, then the parameter value if not empty.
- New parameter: `id.warnings`. If non-zero, warnings are generated for titled objects that don't have titles. True by default; I wonder if this will be too aggressive?
- Added new parameter `keep.relative.image.uris`
- Support default `label` width parameters for itemized and ordered lists
- Added parameters to localize punctuation in indexes.
- Added two new parameters for handling of multi-term `varlistentry` elements:

```
variablelist.term.break.after:
```

When the `variablelist.term.break.after` is non-zero, it will

generate a line break after each `term` multi-term `varlistentry`.

variablelist.term.separator:

When a `varlistentry` contains multiple `term` elements, the string specified in the value of the *variablelist.term.separator* parameter is placed after each `term` except the last. The default is `,` (a comma followed by a space). To suppress rendering of the separator, set the value of *variablelist.term.separator* to the empty string (`""`).

These parameters are primarily intended to be useful if you have multi-term `varlistentries` that have long terms.

Closes #1306676. Thanks to Sam Steingold for providing an example "lots of long terms" doc that demonstrated the value of having these options.

Also, added `normalize-space()` call to processing of each `term`.

This change affects all output formats (HTML, PDF, manpages). The default behavior should pretty much remain the same as before, but it is possible (as always) that the change may introduce some new bugginess.

- Convert 'no' to string in default value.
- Implemented RFE #1292615.

Added bunch of new parameters (attribute sets) that affect list presentation: `list.block.properties`, `itemizedlist.properties`, `orderedlist.properties`, `itemizedlist.label.properties` and `orderedlist.label.properties`. Default behaviour of stylesheets has not been changed but further customizations will be much more easier.

- Implemented RFE #1242092.

You can enable crop marks in your document by setting `crop.marks=1` and `xep.extensions=1`.

Appearance of crop marks can be controlled by parameters `crop.mark.bleed` (6pt), `crop.mark.offset` (24pt) and `crop.mark.width` (0.5pt).

Also there is new named template called `user-xep-pis`. You can overwrite it in order to produce some PIs that can control XEP as described in http://www.renderx.com/reference.html#Output_Formats

- Changed short descriptions in doc for `*autolabel*` params to match new `autolabel` behavior.

1.7 Profiling

The following changes have been made to the `profiling` code since the 1.69.1 release.

- Profiling now works together with namespace stripping (V5 documents). Namespace stripping should work with all stylesheets named `profile-`, even if they are not supporting namespace stripping in a non-profiling variant.
- Moved profiling stage out of templates. This make possible to reuse profiled content by several templates and still maintaing node indentity (needed for example for HTML Help where content is processed multiple times).

I don't know why this was not on the top level before. Maybe some XSLT processors choked on it. I hope this will be OK now.

1.8 Tools

The following changes have been made to the `tools` code since the 1.69.1 release.

- Moved `Makefile.DocBook` from `contrib` module to `xsl` module.

1.9 WordML

The following changes have been made to the `wordml` code since the 1.69.1 release.

- added `contrib` element, better handling of default paragraph style
- added `bridgehead`
- added blocks stylesheet to support bibliographies, glossaries and qandaset
- added `mediaobject` caption
- added callouts
- added Word template file
- added `abstract`, fixed `itemizedlist`, `ulink`
- fixed `Makefile`
added many features to Pages support
added `revhistory`, `inlines`, `highlights`, `abstract`
- fixed handling linebreaks when generating WordML
added Apple Pages support

2 Release 1.69.1

This release is a minor bug-fix update to the 1.69.0 release. Along with bug fixes, it includes one configuration-parameter change: The default value of the `annotation.support` parameter is now 0 (off). The reason for that change is that there have been reports that `annotation` handling is causing a significant performance degradation in processing of large documents with `xsltproc`.

3 Release 1.69.0

The release includes major feature changes, particularly in the **manpages stylesheets**, as well as a large number of bug fixes.

As with all DocBook Project ‘dot zero’ releases, this is an **experimental release**.

3.1 Common

- This release adds localizations for the following languages: Albanian, Amharic, Azerbaijani, Hindi, Irish (Gaelic), Gujarati, Kannada, Mongolian, Oriya, Punjabi, Tagalog, Tamil, Welsh.
 - Added support for specifying number format for auto labels for `chapter`, `appendix`, `part`, and `preface`. Contolled with the `appendix.autolabel`, `chapter.autolabel`, `part.autolabel`, and `preface.autolabel` parameters.
 - Added basic support for `biblioref` cross referencing.
 - Added support for align on `caption` in `mediaobject`.
-

- Added support for processing documents that use the DocBook V5 namespace.
- Added support for `termdef` and `mathphrase`.
- **EXPERIMENTAL:** Incorporated the Slides and Website stylesheets into the DocBook XSL stylesheets package. So, for example, Website documents can now be processed using the following URI for the driver Website `tabular.xsl` file:

`http://docbook.sourceforge.net/release/xsl/current/website/tabular.xsl`

- A procedure without a `title` is now treated as an ‘informal’ procedure (meaning that it is not added to any generated ‘list of procedures’ and has no affect on numbering of generated labels for other procedures).
- `docname` is no longer added to `olink` when pointing to a root element.
- Added support for generation of choice separator in inline `simplelist`. This enables auto-generation of an appropriate localized ‘choice separator’ (for example, ‘and’ or ‘or’) before the final item in an inline `simplelist`.

To indicate that you want a choice separator generated for a particular list, you need to put a processing instruction (PI) of the form `<?dbchoice choice="foo"?>` as a child of the list. For example:

```
<para>Choose from
ONE and ONLY ONE of the following:
<simplelist type="inline">
<?dbchoice choice="or" ?>
<member>A</member>
<member>B</member>
<member>C</member>.</simplelist></para>
```

Output (for English):

Choose from ONE and only ONE of the following choices: A, B, or C.

As a temporary workaround for the fact that most of the DocBook non-English locale files don’t have a localization for the word ‘or’, you can put in a literal string to be used; example for French: `<?dbchoice choice="ou"?>`. That is, use ‘ou’ instead of ‘or’.

3.2 FO

- Added `content-type` property to `external-graphic` element, based on `imagedata` format attribute.
- Added support for generating `<rx:meta-field creator="$VERSION"/>` field for XEP output. This makes the DocBook XSL stylesheet version information available through the Document Properties menu in Acrobat Reader and other PDF viewers.
- Trademark symbol handling made consistent with handling of same in HTML stylesheets. Prior to this change, if you processed a document that contained no value for the class attribute on the `trademark` element, the HTML stylesheets would default to rendering a superscript TM symbol after the `trademark` contents, but the FO stylesheets would render nothing.
- Added support for generating XEP bookmarks for `refentry`.
- Added support for HTML markup `table border` attribute, applied to each table cell.
- The `table.width` template can now sum column specs if none use `%` or `*`.
- Added `fox:destination` extension inside `fox:outline` to support linking to internal destinations.
- Added support for customizing `abstract` with property sets. Controlled with the `abstract.properties` and `abstract-title.properties` parameters.
- Add footnotes in table title to table footnote set, and add support for table footnotes to HTML table markup.
- Added support for `title` in `glosslist`.

- Added support for `itemizedlist` symbol `none`.
- Implemented the new `graphical.admonition.properties` and `nongraphical.admonition.properties` attribute sets.
- Added `id` to `formalpara` and some other blocks that were missing it.
- Changed the anchor template to output `fo:inline` instead of `fo:wrapper`.
- Added support for `toc.max.depth` parameter.

3.3 Help

- Eclipse Help: Added support for generating olink database.

3.4 HTML

- Added a first cut at support in HTML output for DocBook 5 style annotations. Controlled using the `annotation.support` parameter, and implemented using JavaScript and CSS styling. For more details, see the documentation for the `annotation.js`, `annotation.css`, `annotation.graphic.open`, and `annotation.graphic.close` parameters.
- Generate client-side image map for `imageobjectco` with areas using `calspair` units
- Added support for `<?img.src.path?>` PI.
- Added support for passing `img.src.path` to DocBook Java XSLT image extensions when appropriate. Controlled using the `graphicsize.use.img.src.path` parameter.
- Added support for (not valid for DocBook 4) `xlink:href` on `area` and (not valid for DocBook 4) `alt` in `area`.
- Added new parameter `default.table.frame` to control table framing if there is no `frame` attribute on a table.
- Added initial, experimental support for generating content for the HTML `title` attribute from content of the `alt` element. This change adds support for the following inline elements only (none of them are block elements): `abbrev`, `accel`, `acronym`, `action`, `application`, `authorinitials`, `beginpage`, `citation`, `citerefentry`, `citetitle`, `city`, `classname`, `code`, `command`, `computeroutput`, `constant`, `country`, `database`, `email`, `envar`, `errorcode`, `errorname`, `errortext`, `errortype`, `exceptionname`, `fax`, `filename`, `firstname`, `firstterm`, `foreignphrase`, `function`, `glossterm`, `guibutton`, `guicon`, `guilabel`, `guimenu`, `guimenuitem`, `guisubmenu`, `hardware`, `honorific`, `interface`, `interfacename`, `keycap`, `keycode`, `keysym`, `lineage`, `lineannotation`, `literal`, `markup`, `medialabel`, `methodname`, `mousebutton`, `option`, `optional`, `otheraddr`, `othername`, `package`, `parameter`, `personname`, `phone`, `pob`, `postcode`, `productname`, `productnumber`, `prompt`, `property`, `quote`, `refentrytitle`, `remark`, `replaceable`, `returnvalue`, `tag`, `shortcut`, `state`, `street`, `structfield`, `structname`, `subscript`, `superscript`, `surname`, `symbol`, `systemitem`, `tag`, `termdef`, `token`, `trademark`, `type`, `uri`, `userinput`, `varname`, `wordasword`
- Added support for chunking `revhistory` into separate file (similar to the support for doing same with `legalnotice`). Patch from Thomas Schraitle. Controlled through new `generate.revhistory.link` parameter.
- `l10n.xsl`: Made language codes RFC compliant. Added a new boolean config parameter, `l10n.lang.value.rfc.compliant`. If it is non-zero (the default), any underscore in a language code will be converted to a hyphen in HTML output. If it is zero, the language code will be left as-is.

3.5 man

This release closes out 44 manpages stylesheet bug reports and feature requests. It adds more than 35 new configuration parameters for controlling aspects of man-page output -- including hyphenation and justification, handling of links, conversion of Unicode characters, and contents of man-page headers and footers.

- New options for globally disabling/enabling hyphenation and justification: `man.justify` and `man.hyphenate`.

Note that the default for the both of those is zero (off), because justified text looks good only when it is also hyphenated; to quote the ‘Hyphenation’ node from the groff info page:

Since the odds are not great for finding a set of words, for every output line, which fit nicely on a line without inserting excessive amounts of space between words, ‘gtroff’ hyphenates words so that it can justify lines without inserting too much space between words.

The problem is that groff can end up hyphenating a lot of things that you don’t want hyphenated (variable names and command names, for example). Keeping both justification and hyphenation disabled ensures that hyphens won’t get inserted where you don’t want to them, and you don’t end up with lines containing excessive amounts of space between words. These default settings run counter to how most existing man pages are formatted. But there are some notable exceptions, such as the perl man pages.

- Added parameters for controlling hyphenation of computer inlines, filenames, and URLs. By default, even when hyphenation is enabled (globally), hyphenation is now suppressed for "computer inlines" (currently, just `classname`, `constant`, `envar`, `errorcode`, `option`, `replaceable`, `userinput`, `type`, and `varname`, and for filenames, and for URLs from `link`. It can be (re)enabled using the `man.hyphenate.computer.inlines`, `man.hyphenate.filenames`, and `man.hyphenate.urls` parameters.
- Implemented a new system for replacing Unicode characters. There are two parts to the new system: a ‘string substitution map’ for doing ‘essential’ replacements, and a ‘character map’ that can optionally be disabled and enabled.

The new system fixes all open bugs that had to do with literal Unicode numbered entities such as “ and ” showing up in output, and greatly expands the ability of the stylesheets to generate ‘good’ roff equivalents for Unicode symbols and special characters.

Here are some details...

The previous manpages mechanism for replacing Unicode symbols and special characters with roff equivalents (the `replace-entities` template) was not scalable and not complete. The mechanism handled a somewhat arbitrary selection of less than 20 or so Unicode characters. But there are potentially more than 800 Unicode special characters that have some groff equivalent they can be mapped to. And there are about 34 symbols in the Latin-1 (ISO-8859-1) block alone. Users might reasonably expect that if they include any of those Latin-1 characters in their DocBook source documents, they will get correctly converted to known roff equivalents in output.

In addition to those common symbols, certain users may have a need to use symbols from other Unicode blocks. Say, somebody who is documenting an application related to math might need to use a bunch of symbols from the ‘Mathematical Operators’ Unicode block (there are about 65 characters in that block that have reasonable roff equivalents). Or somebody else might really like Dingbats -- such as the checkmark character -- and so might use a bunch of things from the ‘Dingbat’ block (141 characters in that that have roff equivalents or that can at least be ‘degraded’ somewhat gracefully into roff).

So, the old `replace-entities` mechanism was replaced with a completely different mechanism that is based on use of two ‘maps’: a ‘substitution map’ and a ‘character map’ (the latter in a format compliant with the XSLT 2.0 spec and therefore completely ‘forward compatible’ with XSLT 2.0).

The substitution map is controlled through the `man.string.subst.map` parameter, and is used to replace things like the backslash character (which needs special handling to prevent it from being interpreted as a roff escape). The substitution map cannot be disabled, because disabling it will cause the output to be broken. However, you can add to it and change it if needed.

The ‘character map’ mechanism, on the other hand, can be completely disabled. It is enabled by default, and, by default, does replacement of all Latin-1 symbols, along with most special spaces, dashes, and quotes (about 75 characters by default). Also, you can optionally enable a ‘full’ character map that provides support for converting all 800 or so of the characters that have some reasonable groff equivalent.

The character-map mechanism is controlled through the following parameters:

`man.charmap.enabled`

turns character-map support on/off

`man.charmap.use.subset`

specifies that a subset of the character map is used instead of the full map

`man.charmap.subset.profile`

specifies profile of character-map subset

man.charmap.uri

specifies an alternate character map to use instead of the ‘standard’ character map provided in the distribution

- Implemented out-of-line handling of display of URLs for links (currently, only for `ulink`). This gives you three choices for handling of links:
 1. Number and list links. Each link is numbered inline, with a number in square brackets preceding the link contents, and a numbered list of all links is added to the end of the document.
 2. Only list links. Links are not numbered, but an (unnumbered) list of links is added to the end of the document.
 3. Suppress links. Don’t number links and don’t add any list of links to the end of the document.

You can also choose whether links should be underlined. The default is ‘the works’ -- list, number, and underline links. You can use the `man.links.list.enabled`, `man.links.are.numbered`, and `man.links.are.underlined` parameters to change the defaults. The default heading for the link list is REFERENCES. You can change that using the `man.links.list.heading` parameter.

- Changed default output encoding to UTF-8. **This does not mean that man pages are output in raw UTF-8**, because the character map is applied before final output, causing all UTF-8 characters covered in the map to be converted to roff equivalents.
- Added support for processing `refsect3` and `formalpara` and nested `refsection` elements, down to any arbitrary level of nesting.
- Output of the NAME and SYNOPSIS and AUTHOR headings and the headings for admonitions (note, caution, etc.) are no longer hard-coded for English. Instead, headings are generated for those in the correct locale (just as the FO and HTML stylesheets do).
- Re-worked mechanism for assembling page headers/footers (the contents of the `.TH` macro ‘title line’).

Here are some details...

All man pages contain a `.TH` roff macro whose contents are used for rendering the ‘title line’ displayed in the header and footer of each page. Here are a couple of examples of real-world man pages that have useful page headers/footers:

```
gtk-options(7)      GTK+ User's Manual      gtk-options(7) <-- header
GTK+ 1.2            2003-10-20            gtk-options(7) <-- footer

svglib(7)           Svglib User Manual      svglib(7) <-- header
Svglib 1.4.1        16 December 1999        svglib(7) <-- footer
```

And here are the terms with which the `groff_man(7)` man page refers to the various parts of the header/footer:

```
title(section)  extra3  title(section)  <- header
extra2          extra1  title(section)  <- footer
```

Or, using the names with which the `man(7)` man page refers to those same fields:

```
title(section)  manual  title(section)  <- page header
source          date    title(section)  <- page footer
```

The easiest way to control the contents of those fields is to mark up your `refentry` content like the following (note that this is a ‘minimal’ example).

```
<refentry>
<info>
  <date>2003-10-20</date> ❶
</info>
<refmeta>
  <refentrytitle>gtk-options</refentrytitle> ❷
  <manvolnum>7</manvolnum> ❸
  <refmiscinfo class="source-name">GTK+</refmiscinfo> ❹
  <refmiscinfo class="version">1.2</refmiscinfo> ❺
```

```

    <refmiscinfo class="manual">GTK+ User's Manual</refmiscinfo> ❹
  </refmeta>
  <refnamediv>
    <refname>gtk-options</refname>
    <refpurpose>Standard Command Line Options for GTK+ Programs</refpurpose>
  </refnamediv>
  <refsect1>
    <title>Description</title>
    <para>This manual page describes the command line options, which
      are common to all GTK+ based applications.</para>
  </refsect1>
</refentry>

```

- ❶ Sets the ‘date’ part of the header/footer.
- ❷ Sets the ‘title’ part.
- ❸ Sets the ‘section’ part.
- ❹ Sets the ‘source name’ part.
- ❺ Sets the ‘version’ part.
- ❻ Sets the ‘manual’ part.

Below are explanations of the steps the stylesheets take to attempt to assemble and display ‘good’ headers and footer. [In the descriptions, note that **info* is the *refentry* ‘info’ child (whatever its name), and *parentinfo* is the ‘info’ child of its parent (again, whatever its name).]

extra1 field (date)

Content of the ‘extra1’ field is what shows up in the **center footer** position of each page. The `man(7)` man page describes it as ‘the date of the last revision’.

To provide this content, if the `refentry.date.profile.enabled` is non-zero, the stylesheets check the value of `refentry.date.profile`.

Otherwise, by default, they check for a date or pubdate not only in the **info* contents, but also in the *parentinfo* contents.

If a date cannot be found, the stylesheets now automatically generate a localized ‘long format’ date, ensuring that this field always has content in output.

However, if for some reason you want to suppress this field, you can do so by setting a non-zero value for `man.th.extra1.suppress`.

extra2 field (source)

On Linux systems and on systems with a modern groff, the content of the ‘extra2’ field are what shows up in the **left footer** position of each page.

The `man(7)` man page describes this as ‘the source of the command’, and provides the following examples:

- For binaries, use something like: GNU, NET-2, SLS Distribution, MCC Distribution.
- For system calls, use the version of the kernel that you are currently looking at: Linux 0.99.11.
- For library calls, use the source of the function: GNU, BSD 4.3, Linux DLL 4.4.1.

In practice, there are many pages that simply have a version number in the ‘source’ field. So, it looks like what we have is a two-part field, *Name Version*, where:

Name

product name (e.g., BSD) or org. name (e.g., GNU)

Version

version name

Each part is optional. If the *Name* is a product name, then the *Version* is probably the version of the product. Or there may be no *Name*, in which case, if there is a *Version*, it is probably the version of the item itself, not the product it is part of. Or, if the *Name* is an organization name, then there probably will be no *Version*.

To provide this content, if the `refentry.source.name.profile.enabled` and `refentry.version.profile.enabled` parameter are non-zero, the stylesheets check the value of `refentry.source.name.profile` `refentry.version.profile`.

Otherwise, by default, they check the following places, in the following order:

1. `*info/productnumber`
2. `*info/productnumber`
3. `refmeta/refmiscinfo[@class = 'version']`
4. `parentinfo/productnumber`
5. `*info/productname`
6. `parentinfo/productname`
7. `refmeta/refmiscinfo`
8. [nothing found, so leave it empty]

extra3 field

On Linux systems and on systems with a modern groff, the content of the 'extra3' field are what shows up in the **center header** position of each page. Some man pages have 'extra2' content, some don't. If a particular man page has it, it is most often 'context' data about some larger system the documented item belongs to (for example, the name or description of a group of related applications). The stylesheets now check the following places, in the following order, to look for content to add to the 'extra3' field.

1. `parentinfo/title`
2. parent's title
3. `refmeta/refmiscinfo`
4. [nothing found, so leave it empty]

- Reworked `*info` gathering. For each `refentry` found, the stylesheets now cache its `*info` content, then check for any valid parent of it that might have `metainfo` content and cache that, if found; they then then do all further matches against those node-sets (rather than re-selecting the original `*info` nodes each time they are needed).
- New option for breaking strings after forward slashes. This enables long URLs and pathnames to be broken across lines. Controlled through `man.break.after.slash` parameter.
- Output for servicemark and trademark are now (SM) and (TM). There is a groff `"\ (tm"` escape, but output from that is not acceptable.
- New option for controlling the length of the title part of the .TH title line. Controlled through the `man.th.title.max.length` parameter.
- New option for specifying output encoding of each man page; controlled with `man.output.encoding` (similar to the HTML `chunker.output.encoding` parameter).
- New option for suppressing filename messages when generating output; controlled with `man.output.quietly` (similar to the HTML `chunk.quietly` parameter).
- The text of cross-references to first-level `refentry` (`refsect1`, top-level `refsection`, `refnamediv`, and `refsynopsisdiv`) are now capitalized.
- Cross-references to `refnamediv` now use the localized NAME title instead of using the first `refname` child. This makes the output inconsistent with HTML and FO output, but for man-page output, it seems to make better sense to have the NAME. (It may actually make better sense to do it that way in HTML and FO output as well...)
- Added support for processing `funcparams`.
- Removed the space that was being output between `funcdef` and `paramdef`; example: was: `float rand (void)`; now: `float rand(void)`
- Turned off bold formatting for the `type` element when it occurs within a `funcdef` or `paramdef`
- Corrected rendering of `simplelist`. Any `<simplelist type="inline"` instance is now rendered as a comma-separated list (also with an optional localized 'and' or 'or' before the last item -- see description elsewhere in these release notes). Any `simplelist` instance whose `type` is not `inline` is rendered as a one-column vertical list (ignoring the values of the `type` and `columns` attributes if present)
- Comment added at top of roff source for each page now includes DocBook XSL stylesheets version number (as in the HTML stylesheets)

- Made change to prevent ‘sticky’ fonts changes. Now, when the manpages stylesheets encounter node sets that need to be boldfaced or italicized, they put the `\fBfoo\fR` and `\fIbar\fR` groff bold/italic instructions separately around each node in the set.

- `synop.xml`: Boldface everything in `funcsynopsis` output except parameters (which are in ital). The `man(7)` man page says:

For functions, the arguments are always specified using italics, even in the SYNOPSIS section, where the rest of the function is specified in bold.

A look through the contents of the `man/man2` directory shows that most (all) existing pages do follow this ‘everything in `funcsynopsis` bold’ rule. That means the `type` content and any punctuation (parens, semicolons, `varargs`) also must be bolded.

- Removed code for adding backslashes before periods/dots in roff source, because backslashes in front of periods/dots in roff source are needed only in the very rare case where a period is the very first character in a line, without any space in front of it. A better way to deal with that rare case is for you to add a zero-width space in front of the offending dot(s) in your source
- Removed special handling of the `quote` element. That was hard-coded to cause anything marked up with the `quote` element to be output preceded by two backticks and followed by two apostrophes -- that is, that old-school kludge for generating ‘curly’ quotes in Emacs and in X-Windows fonts. While Emacs still seems to support that, I don’t think X-Windows has for a long time now. And, anyway, it looks (and has always looked) like crap when viewed on a normal tty/console. In addition, it breaks localization of `quote`. By default, `quote` content is output with localized quotation marks, which, depending on the locale, may or may not be left and right double quotation marks.
- Changed mappings for left and right single quotation marks. Those had previously been incorrectly mapped to the backtick (```) and apostrophe (`&39;`) characters (for kludgy reasons -- see above). They are now correctly mapped to the `\(oq` and `\(cq` roff escapes. If you want the old (broken) behavior, you need to manually change the mappings for those in the value of the `man.string.subst.map` parameter.
- Removed `xref.xml` file. Now, of the various cross-reference elements, only the `ulink` element is handled differently; the rest are handled exactly as the HTML stylesheets handle them, except that no hypertext links are generated. (Because there is no equivalent hypertext mechanism in man pages.)
- New option for making ‘subheading dividers’ in generated roff source. The dividers are not visible in the rendered man page; they are just there to make the source readable. Controlled using `man.subheading.divider`.
- Fixed many places where too much space was being added between lines.

4 Release 1.68.1

The release adds localization support for Farsi (thanks to Sina Heshmati) and improved support for the XLink-based DocBook NG `db:link` element. Other than that, it is a minor bug-fix update to the 1.68.0 release. The main thing it fixes is a build error that caused the XSLT Java extensions to be jarred up with the wrong package structure. Thanks to Jens Stavnstrup for quickly reporting the problem, and to Mauritz Jeanson for investigating and finding the cause.

5 Release 1.68.0

This release includes some features changes, particularly for FO/PDF output, and a number of bug fixes.

FO

- Moved footnote properties to attribute-sets.
- Added support for side floats, margin notes, and custom floats.
- Added new parameters `body.start.indent` and `body.end.indent` to the `set.flow.properties` template.
- Added support for `xml:id`

- Added support for `refdescriptor`.
- Added support for multiple `refnamedivs`.
- Added `index.entry.properties` attribute-set to support customization of index entries.
- Added `set.flow.properties` `template` call to each `fo:flow` to support customizations entry point.
- Add support for `@floatstyle` in figure
- Moved hardcoded properties for index division titles to the `index.div.title.properties` attribute-set.
- Added support for `table-layout="auto"` for XEP.
- Added `index.div.title.properties` attribute-set.
- `$verbose` parameter is now passed to most elements.
- Added `refentry` to `toc` in part, as it is permitted by the DocBook schema/DTD.
- Added backmatter elements and `article` to `toc` in part, since they are permitted by the DocBook schema/DTD.
- Added `mode="toc"` for `simplesect`, since it is now permitted in the `toc` if `simplesect.in.toc` is set.
- Moved hard-coded properties to `nongraphical.admonition.properties` and `graphical.admonition.properties` attribute sets.
- Added support for `sidebar-width` and `float-type` processing instructions in sidebar.
- For tables with HTML markup elements, added support for `dbfo` `bgcolor` PI, the attribute-sets named `table.properties`, `informaltable.properties`, `table.table.properties`, and `table.cell.padding`. Also added support for the templates named `table.cell.properties` and `table.cell.block.properties` so that `tabstyles` can be implemented. Also added support for tables containing only `tr` instead of `tbody` with `tr`.
- Added new parameter `hyphenate.verbatim.characters` which can specify characters after which a line break can occur in verbatim environments. This parameter can be used to extend the initial set of characters which contain only space and non-breakable space.
- Added `itemizedlist.label.markup` to enable selection of different bullet symbol. Also added several potential bullet characters, commented out by default.
- Enabled all id's in XEP output for external olinking.

HTML

- Added support for `refdescriptor`.
- Added support for multiple `refnamedivs`.
- Added support for `xml:id`
- `refsynopsisdiv` as a section for counting section levels

IMAGES

- Added new SVG admonition graphics and navigation images.

6 Release 1.67.2

This release fixes a table bug introduced in the 1.67.1 release.

7 Release 1.67.1

This release includes a number of bug fixes.

The following lists provide details about API and feature changes.

FO

- Tables: Inherited cell properties are now passed to the `table.cell.properties` template so they can be overridden by a customization.
- Tables: Added support for bgcolor PI on table row element.
- TOCs: Added new parameter `simplesect.in.toc`; default value of 0 causes simplesect to be omitted from TOCs; to cause simplesect to be included in TOCs, you must set the value of `simplesect.in.toc` to 1. Comment from Norm:

Simplesect elements aren't supposed to appear in the ToC at all... The use case for simplesect is when, for example, every chapter in a book ends with "Exercises" or "For More Information" sections and you don't want those to appear in the ToC.

- Sections: Reverted change that caused a variable reference to be used in a template match and rewrote code to preserve intended semantics.
- Lists: Added workaround to prevent "`* 0.60 + 1em`" garbage in list output from PassiveTeX
- Moved the literal attributes from `component.title` to the `component.title.properties` attribute-set so they can be customized.
- Lists: Added glossdef's first para to special handling in `fo:list-item-body`.

HTML

- TOCs: Added new parameter `simplesect.in.toc`; for details, see the list of [?] changes for this release.
- Indexing: Added new parameter `index.prefer.titleabbrev`; when set to 1, index references will use `titleabbrev` instead of `title` when available.

HTML HELP

- Added support for generating windows-1252-encoded output using Saxon; for more details, see the list of [?] changes for this release.

MAN PAGES

- Replaced named/numeric character-entity references for non-breaking space with groff equivalent (backslash-tilde).

XSL JAVA EXTENSIONS

- Saxon extensions: Added the `Windows1252` class. It extends Saxon 6.5.x with the windows-1252 character set, which is particularly useful when generating HTML Help for Western European Languages (code from Pontus Haglund and contributed to the DocBook community by Sectra AB, Sweden).

To use:

1. Make sure that the Saxon 6.5.x jar file and the jar file for the DocBook XSL Java extensions are in your `CLASSPATH`
2. Create a DocBook XSL customization layer -- a file named `mystylesheet.xsl` or whatever -- that, at a minimum, contains the following:

```
<xsl:stylesheet
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  version='1.0'>
  <xsl:import href="http://docbook.sourceforge.net/release/xsl/current/htmlhelp/ ↵
    htmlhelp.xsl"/>
  <xsl:output method="html" encoding="WINDOWS-1252" indent="no"/>
  <xsl:param name="htmlhelp.encoding" select="'WINDOWS-1252'"/></xsl:param>
  <xsl:param name="chunker.output.encoding" select="'WINDOWS-1252'"/></xsl:param>
  <xsl:param name="saxon.character.representation" select="'native'"/></xsl:param>
</xsl:stylesheet>
```

Invoke Saxon with the `encoding.windows-1252` Java system property set to `com.nwalsh.saxon.Windows1252`; for example

```
java \
  -Dencoding.windows-1252=com.nwalsh.saxon.Windows1252 \
  com.icl.saxon.StyleSheet \
  mydoc.xml mystylesheet.xsl
```

Or, for a more complete "real world" case showing other options you'll typically want to use:

```
java \
  -Dencoding.windows-1252=com.nwalsh.saxon.Windows1252 \
  -Djavax.xml.parsers.DocumentBuilderFactory=org.apache.xerces.jaxp. ↵
    DocumentBuilderFactoryImpl \
  -Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl \
  -Djavax.xml.transform.TransformerFactory=com.icl.saxon.TransformerFactoryImpl \
  com.icl.saxon.StyleSheet \
  -x org.apache.xml.resolver.tools.ResolvingXMLReader \
  -y org.apache.xml.resolver.tools.ResolvingXMLReader \
  -r org.apache.xml.resolver.tools.CatalogResolver \
  mydoc.xml mystylesheet.xsl
```

In both cases, the "mystylesheet.xml" file should be a DocBook customization layer containing the parameters show in step 2.

- Saxon extensions: Removed Saxon 8 extensions from release package

8 Release 1.67.0

- A number of important bug fixes.
- Added Saxon8 extensions
- Enabled `dbfo table-width` on `entrytbl` in FO output
- Added support for `role=strong` on `emphasis` in FO output
- Added new FO parameter `hyphenate.verbatim` that can be used to turn on "intelligent" wrapping of verbatim environments.
- Replaced all `<tt></tt>` output with `<code></code>`
- Changed `admon.graphic.width` template to a mode so that different admonitions can have different graphical widths.
- Deprecated the HTML `shade.verbatim` parameter (use CSS instead)
- Wrapped ToC `refentrytitle/refname` and `refpurpose` in span with class values. This makes it possible to style them using a CSS stylesheet.
- Use `strong/em` instead of `b/i` in HTML output

- Added support for converting `Emphasis` to groff italic and `Emphasis role='bold'` to bold. Controlled by `emphasis.-propagates.style` param, but not documented yet using litprog system. Will do that next (planning to add some other parameter-controllable options for hyphenation and handling of line spacing).
- `callout.graphics.number.limit.xml` param: Changed the default from 10 to 15.
- `verbatim.properties`: Added `hyphenate=false`
- Saxon and Xalan Text.java extensions: Added support for `URIResolver()` on `insertfile href`'s
- Added generated `RELEASE-NOTES.txt` file.
- Added `INSTALL` file (executable file for generating `catalog.xml`)
- Removed obsolete `tools` directory from package

9 Release 1.66.1

- A number of important bug fixes.
- Now `xml:base` attributes that are generated by an XInclude processor are resolved for image files.
- Rewrote olink templates to support several new features.
 - Extended full olink support to FO output.
 - Add support for `xrefstyle` attribute in olinks.
 - New parameters to support new olink features: `insert.olink.page.number`, `insert.olink.pdf.frag`, `olink.debug`, `olink.lang.fallback.sequence`, `olink.properties`, `prefer.internal.olink`. See the reference page for each parameter for more information.
- Added `index.on.type` parameter for new type attribute introduced in DocBook 4.3 for `indexterm`s and `index`. This allows you to create multiple indices containing different categories of entries. For users of 4.2 and earlier, you can use the new parameter `index.on.role` instead.
- Added new `section.autolabel.max.depth` parameter to turn off section numbering below a certain depth. This permits you to number major section levels and leave minor section levels unnumbered.
- Added `footnote.sep.leader.properties` attribute set to format the line separating footnotes in printed output.
- Added parameter `img.src.path` as a prefix to HTML `img src` attributes. The prefix is added to whatever path is already generated by the stylesheet for each image file.
- Added new attribute-sets `informalequation.properties`, `informalexample.properties`, `informalfigure.properties`, and `informaltable.properties`, so each such element type can be formatted individually if needed.
- Add `component.label.includes.part.label` parameter to add any part number to chapter, appendix and other component labels when the `label.from.part` parameter is nonzero. This permits you to distinguish multiple chapters with the same chapter number in cross references and the TOC.
- Added `chunk.separate.lots` parameter for HTML output. This parameter lets you generate separate chunk files for each LOT (list of tables, list of figures, etc.).
- Added several table features:
 - Added `table.table.properties` attribute set to add properties to the `fo:table` element.
 - Added placeholder templates named `table.cell.properties` and `table.cell.block.properties` to enable adding properties to any `fo:table-cell` or the cell's `fo:block`, respectively. These templates are a start for implementing table styles.
- Added new attribute set `component.title.properties` for easy modifications of component's title formatting in FO output.

-
- Added Saxon support for an encoding attribute on the `textdata` element. Added new parameter `textdata.default.encoding` which specifies encoding when encoding attribute on `textdata` is missing.
 - Template `label.this.section` now controls whole section label, not only sub-label which corresponds to particular label. Former behaviour was IMHO bug as it was not usable.
 - Formatting in `titleabbrev` for TOC and headers is preserved when there are no hotlink elements in the title. Formerly the title showed only the text of the title, no font changes or other markup.
 - Added `intial.page.number` template to set the `initial-page-number` property for page sequences in print output. Customizing this template lets you change when page numbering restarts. This is similar to the `format.page.number` template that lets you change how the page number formatting changes in the output.
 - Added `force.page.count` template to set the `force-page-count` property for page sequences in print output. This is similar to the `format.page.number` template.
 - Sort language for localized index sorting in `autoidx-ng.xml` is now taken from document `lang`, not from system environment.
 - Numbering and formatting of normal and ulink footnotes (if turned on) has been unified. Now ulink footnotes are mixed in with any other footnotes.
 - Added support for `renderas` attribute in `section` and `sect1` et al. This permits you to render a given section title as if it were a different level.
 - Added support for `label` attribute in footnote to manually supply the footnote mark.
 - Added support for DocBook 4.3 `corpcredit` element.
 - Added support for a `dbfo keep-together` PI for formal objects (table, figure, example, equation, programlisting). That permits a formal object to be kept together if it is not already, or to be broken if it is very long and the default keep-together is not appropriate.
 - For graphics files, made file extension matching case insensitive, and updated the list of graphics extensions.
 - Allow `calloutlist` to have block content before the first callout
 - Added `dbfo-need` processing instruction to provide soft page breaks.
 - Added implementation of existing but unused `default.image.width` parameter for graphics.
 - Support DocBook NG `tag` inline element.
 - It appears that XEP now supports Unicode characters in bookmarks. There is no further need to strip accents from characters.
 - Make `segmentedlist` HTML markup more semantic and available to CSS styles.
 - Added `user.preroot` placeholder template to permit `xsl-stylesheet` and other PIs and comments to be output before the HTML root element.
 - Non-chunked `legalnotice` now gets an `` element in HTML output so it can be referenced with `xref` or `link`.
 - In chunked HTML output, changed `link rel="home"` to `rel="start"`, and `link rel="previous"` to `rel="prev"`, per W3C HTML 4.01 spec.
 - Added several patches to `htmlhelp` from W. Borgert
 - Added Bosnian locale file as `common/bs.xml`.
-

10 Release 1.65.0

- A number of important bug fixes.
- Added a workaround to allow these stylesheets to process DocBook NG documents. (It's a hack that pre-processes the document to strip off the namespace and then uses `exsl:node-set` to process the result.)
- Added alternative indexing mechanism which has better internationalization support. New indexing method allows grouping of accented letters like e, é, ë into the same group under letter "e". It can also treat special letters (e.g. "ch") as one character and place them in the correct position (e.g. between "h" and "i" in Czech language).

In order to use this mechanism you must create customization layer which imports some base stylesheet (like `fo/docbook.xsl`, `html/chunk.xsl`) and then includes appropriate stylesheet with new indexing code (`fo/autoidx-ng.xsl` or `html/autoidx-ng.xsl`). For example:

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

<xsl:import href="http://docbook.sourceforge.net/release/xsl/current/fo/docbook.xsl"/>
<xsl:include href="http://docbook.sourceforge.net/release/xsl/current/fo/autoidx-ng.xsl"/>

</xsl:stylesheet>
```

New method is known to work with Saxon and it should also work with `xsltproc` 1.1.1 and later. Currently supported languages are English, Czech, German, French, Spanish and Danish.

11 Release 1.64.1

- In the course of fixing bug #849787, wrapping Unicode callouts with an appropriate font change in the Xalan extensions, I discovered that the Xalan APIs have changed a bit. So `xalan2.jar` will work with older Xalan 2 implementations, `xalan25.jar` works with Xalan 2.5.

12 Release 1.61.0

- Initial support for timestamp PI. From now you can use `<?dbtimestamp format="Y-m-d H:M:S">` to get current datetime in your document. Added localization support for datetime PI
- Added level 6 to test for `section` depth in `section.level` template so that `section.title.level6.properties` will be used for sections that are 6 deep or deeper. This should also cause a `h6` to be created in html output.
- Don't use SVG graphics if `use.svg=0`
- Now uses `number-and-title-template` for sections only if `section.autolabel` is not zero.
- Added missing 'english-language-name' attribute to the `l10n` element, and the missing 'style' attribute to the template element so the current gentext documents will validate.
- Corrected several references to parameter `qanda.defaultlabel` that were missing the "\$".
- Now accepts `admon.textlabel` parameter to turn off Note, Warning, etc. label.
- FeatReq #684561: support more XEP metadata
- Added hyphenation support. Added support for `coref`. Added `beginpage` support. (does nothing; see TDG).
- Added support for `hyphenation-character`, `hyphenation-push-character-count`, and `hyphenation-remain-character-count`
- Added `root.properties`, `ebnf.assignment`, and `ebnf.statement.terminator`

- Support bgcolor PI in table cells; make sure rowsep and colsep don't have any effect on the last row or column
- Handle othercredit on titlepage a little better
- Applied fix from Jeff Beal that fixed the bug that put secondary page numbers on primary entries. Same with tertiary page numbers on secondary entries.
- Added definition of missing variable *collection*.
- Make footnote formatting 'normal' even when it occurs in a context that has special formatting
- Added warning when *glossary.collection* is not blank, but it cannot open the specified file.
- Pick up the frame attribute on table and informatable.
- indexdiv/title in non-autogenerated indexes are now picked up.
- Removed (unused) component.title.properties
- Move IDs from page-sequences down to titlepage blocks
- Use proportional-column-width(1) on more tables.
Use proportional-column-width() for header/footer tables; suppress relative-align when using FOP
- Check for *glossterm.auto.link* when linking firstterms; don't output gl. prefix on glossterm links
- Generate Part ToCs
- Support glossary, bibliography, and index in component ToCs.
- Refactored chunking code so that customization of chunk algorithm and chunk elements is more practical
- Support textobject/phrase on inlinemediaobject.
- Support 'start' PI on ordered lists
- Fixed test of \$toc PI to turn on qandaset TOC.
- Added process.chunk.footnotes to sect2 through 5 to fix bug of missing footnotes when chunk level greater than 1.
- Added parameter *toc.max.depth* which controls maximal depth of ToC as requested by PHP-DOC group.
- Exempted titleabbrev from preamble processing in lists, and fixed variablelist preamble code to use the same syntax as the other lists.
- Added support for elements between variablelist and first varlistentry since DocBook 4.2 supports that now.

13 Release 1.60.1

- The format of the titlepage.templates.xml files and the stylesheet that transforms them have been significantly changed. All of the attributes used to control the templates are now namespace qualified. So what used to be:

```
<t:titlepage element="article" wrapper="fo:block">
```

is now:

```
<t:titlepage t:element="article" t:wrapper="fo:block">
```

Attributes from other namespaces (including those that are unqualified) are now copied directly through. In practice, this means that the names that used to be 'fo:' qualified:

```
<title named-template="component.title"
  param:node="ancestor-or-self::article[1]"
  fo:text-align="center"
  fo:keep-with-next="always"
  fo:font-size="&hsize5;"
  fo:font-weight="bold"
  fo:font-family="{ $title.font.family }"/>
```

are now unqualified:

```
<title t:named-template="component.title"
  param:node="ancestor-or-self::article[1]"
  text-align="center"
  keep-with-next="always"
  font-size="&hsize5;"
  font-weight="bold"
  font-family="{ $title.font.family }"/>
```

The `t:titlepage` and `t:titlepage-content` elements both generate wrappers now. And unqualified attributes on those elements are passed through. This means that you can now make the title font apply to an entire titlepage and make the entire ‘recto’ titlepage centered by specifying the font and alignment on those elements:

```
<t:titlepage t:element="article" t:wrapper="fo:block"
  font-family="{ $title.font.family }">

  <t:titlepage-content t:side="recto"
    text-align="center">
```

- Support use of `titleabbrev` in running headers and footers.
- Added (experimental) `xref.with.number.and.title` parameter to enable number/title cross references even when the default would be just the number.
- Generate part ToCs if they’re requested.
- Use `proportional-column-width()` in header/footer tables.
- Handle alignment correctly when screenshot wraps a graphic in a figure.
- Format chapter and appendix cross references consistently.
- Attempt to support tables with multiple `tgroups` in FO.
- Output `fo:table-columns` in `simplelist` tables.
- Use `titlepage.templates.xml` for `indexdiv` and `glossdiv` formatting.
- Improve support for new bibliography elements.
- Added `footnote.number.format`, `table.footnote.number.format`, `footnote.number.symbols`, and `table.footnote.number.symbols` for better control of footnote markers.
- Added `glossentry.show.acronyms`.
- Suppress the draft-mode page masters when `draft-mode` is ‘no’.
- Make blank pages verso not recto. D’Oh!
- Improved formatting of `ulink` footnotes.
- Fixed bugs in graphic width/height calculations.
- Added class attributes to inline elements.

- Don't add '.html' to the filenames identified with the 'dbhtml' PI.
- Don't force a ToC when sections contain `refentry`s.
- Make section title sizes a function of the `body.master.size`.

14 Release 1.59.2

- Removed the region-name from the region-body of blank pages. There's no reason to give the body of blank pages a unique name and doing so causes a mismatch that FOP detects.
- Output IDs for the first paragraphs in listitems.
- Fixed some small bugs in the handling of page numbers in double-sided mode.
- Attempt to prevent duplicated IDs from being produced when endterm on `xref` points to something with nested structure.
- Fix alignment problems in `equations`.
- Output the type attribute on unordered lists (UL) in HTML only if the `css.decoration` parameter is true.
- Calculate the font size in `formal.title.properties` so that it's 1.2 times the base font size, not a fixed "12pt".

15 Release 1.59.1

- Added Bulgarian localization.
- Indexing improvements; localize book indexes to books but allow `setindex` to index an entire set.
- The default value for `rowsep` and `colsep` is now "1" as per CALS.
- Added support for `titleabbrev` (use them for cross references).
- Improvements to `mediaobject` for selecting print vs. online images.
- Added separate property sets for `figures`, `examples`, `equations`, `tabless`, and `procedures`.
- Make `lineannotations` italic.
- Support `xrefstyle` attribute.
- Make endterm on `xref` higher priority than `xreflabel` target.
- Glossary formatting improvements.

16 Release 1.58.0

- This release contains the first attempt at extension support for `xsltproc`. The only extension available to date is the one that adjusts table column widths. Run `extensions/xsltproc/python/xslt.py`.
 - Fixed bugs in calculation of adjusted column widths to correct for rounding errors.
 - Support nested `refsection` elements correctly.
 - Reworked `gentext.template` to take context into consideration. The name of elements in localization files is now an xpath-like context list, not just a simple name.
 - Made some improvements to bibliography formatting.
 - Improved graphical formatting of admonitions.
-

- Added support for `entrytbl`.
- Support spanning index terms.
- Support `bibliosource`.

17 Release 1.57.0

- The 1.57.0 release wasn't documented here. Oops.

18 Release 1.56.0

- Reworked chunking. *This will break all existing customizations layers that change the chunking algorithm.* If you're customizing chunking, look at the new 'content' parameter that's passed to `process-chunk-element` and friends.
- Support continued and inherited numeration in `orderedlist` formatting for FOs.
- Added Thai localization.
- Tweaked stylesheet documentation stylesheets to link to TDG and the parameter references.
- Allow title on tables of contents ("Table of Contents") to be optional. Added new keyword to `generate.toc`. Support tables of contents on sections.
- Made separate parameters for table borders and table cell borders: `table.frame.border.color`, `table.frame.border.style`, `table.frame.border.thickness`, `table.cell.border.color`, `table.cell.border.style`, and `table.cell.border.thickness`.
- Suppress formatting of 'endofrange' `indexterms`. This is only half-right. They should generate a range, but I haven't figured out how to do that yet.
- Support `revdescription`. (Bug #582192)
- Added `default.float.class` and fixed figure floats. (Bug #497603)
- Fixed formatting of `sbr` in FOs.
- Added context to the 'missing template' error message.
- Process `arg` correctly in a group. (Bug #605150)
- Removed 'keep-with-next' from `formal.title.properties` attribute set now that the stylesheets support the option of putting such titles below the object. Now the `$placement` value determines if 'keep-with-next' or 'keep-with-previous' is used in the title block.
- Wrap 'url()' around external-destinations when appropriate.
- Fixed typo in compact list spacing. (Bug #615464)
- Removed spurious hash in anchor name. (Bug #617717)
- Address is now displayed verbatim on title pages. (Bug #618600)
- The `bridgehead.in.toc` parameter is now properly supported.
- Improved effectiveness of HTML cleanup by increasing the number of places where it is used. Improve use of HTML cleanup in XHTML stylesheets.
- Support table of contents for appendix in article. (Bug #596599)
- Don't duplicate footnotes in `bibliographys` and `glossarys`. (Bug #583282)

- Added *default.image.width*. (Bug #516859)
- Totally reworked *functsynopsis* code; it now supports a 'tabular' presentation style for 'wide' prototypes; see *functsynopsis.tabular.threshold*. (HTML only right now, I think, FO support, uh, real soon now.)
- Reworked support for difference marking; toned down the colors a bit and added a 'system.head.content' template so that the diff CSS wasn't overriding 'user.head.content'. (Bug #610660)
- Added call to the '*.head.content' elements when writing out long description chunks.
- Make sure *legalnotice* link is correct even when chunking to a different *base.dir*.
- Use CSS to set viewport characteristics if *css.decoration* is non-zero, use div instead of p for making graphic a block element; make figure titles the default alt text for images in a figure.
- Added space-after to *list.block.spacing*.
- Reworked *section.level* template to give 'correct' answer instead of being off by one.
- When processing tables, use the *tabstyle* attribute as the division class.
- Fixed bug in *html2xhtml.xsl* that was causing the XHTML chunker to output HTML instead of XHTML.

19 Older releases

To view the release notes for older releases, see <http://cvs.sourceforge.net/viewcvs.py/docbook/xsl/RELEASE-NOTES.xml>. Be aware that there were no release notes for releases prior to the 1.50.0 release.

20 About dot-zero releases

DocBook Project 'dot zero' releases should be considered *experimental* and are always followed by stable 'dot one' releases, usually within two or three weeks. Please help to ensure the stability of 'dot one' releases by carefully testing each 'dot zero' release and reporting back about any problems you find.

It is not recommended that you use a 'dot zero' release in a production system, or package it for an OS distro. Instead, you should wait for the 'dot one' version.
