
PTXprint

Technical Reference Manual

version 0.97 (Describing features of PTXprint 3.0.32)

Contents

Introduction.....	5
Who this manual is for.....	5
How to use it.....	5
What's in each section.....	5
Related documents.....	5
PTXprint Specific.....	6
Piclists.....	6
Adjustment Lists.....	7
Keyboard Shortcuts.....	7
Z-Codes.....	7
Table of Contents types.....	10
Marker Style Fields (.sty Files).....	11
Sidebar/Box Style Fields.....	12
Stylesheet Processing Order.....	13
Border pattern syntax.....	13
Marker Grammar.....	14
Key Styles.....	14
Headers.....	15
Changes syntax elements.....	15
References.....	15
Other changes file content.....	16
Peripherals.....	16
Page numbering and numbering resets.....	17
Command-line Options.....	17
Logging.....	19
Diglot merge debugging.....	19
File Processing and Generated Files.....	21
Content-defining macros.....	21
Triggers.....	22
Global Settings and Commands.....	23
Global (layout-altering and <i>debugging</i>) booleans.....	23
Global Dimensions and Counts.....	24
Defined Constants.....	26
Other TeX file commands/options.....	28
PTX Penalty Values.....	28
PTX Special Spaces and Characters.....	29
Hooks.....	30
Sethooks.....	30
Other settable hooks.....	30
AddHooks.....	30
Special hook macros.....	31
Status tests / values for hooks etc.....	31
When hooks aren't enough: complex layouts.....	32
Complex markers and the style stack.....	33
Dimension Units.....	33
Layout Versions.....	34
Tracing Codes.....	35
Plugins.....	35
TeX Info.....	36
Tracers.....	36
Horizontal Penalties and Demerits.....	36
Vertical Penalties.....	36
Booknames.....	37
Regexp.....	39

Special Characters.....	39
Escapes.....	40
Flags.....	40
Properties.....	41
Binary Properties.....	42
Character Sets.....	43
Printers.....	44
Pretore.....	44

Introduction

Who this manual is for

This is a *reference* for people working below the PTXprint user interface: macro and stylesheet authors, advanced users writing changes .txt, triggers, hooks, custom \z-codes or PDF-side TeX, and developers maintaining the ptx2pdf macros. It assumes working knowledge of USFM and of TeX/XeTeX concepts (boxes, glue, penalties, modes). It is **not** an introduction to PTXprint; for task-oriented help see the user documentation and the topic notes in docs/documentation/*.md.

How to use it

Entries are grouped by subsystem, not by workflow. Use the table of contents (or your reader's search) to jump to a marker, setting, or macro name. Conventions: \Marker = a stylesheet field or TeX control sequence; *italics* = a value you supply; [...] = optional; a|b = alternatives; default values appear in square brackets at the end of a description, e.g. [*false*]. Names containing @ are internal and require \makeatletter to use directly.

What's in each section

- **PTXprint Specific** — everything that adds on top of plain USFM/TeX.
- **TeX Info** — the underlying XeTeX knobs: tracing switches, horizontal/vertical penalties and demerits.
- **Booknames** — the three-letter / two-letter book ID tables.
- **Regexp** — the regular-expression syntax used in changes .txt and elsewhere (this is the regex module syntax, not POSIX).

Related documents

This manual is deliberately terse — it lists fields, codes, and parameters but rarely explains the reasoning behind them or shows them in use. For that, see the companion topic guides in docs/documentation/, which cover the same subsystems discursively and with worked examples: SideBars.md (the \esb box model), figures.md and ornaments.md (illustrations and ornamental borders/separators), diglot.md and polyglot.md (parallel-column typesetting), covers.md and thumbtabs.md (cover layout and edge tabs), footnotes.md and glue.md (note control and spacing/line-breaking behaviour), styling.md (how styling resolves in complex marker contexts), special_chars.md (Unicode layout-control characters), Triggers.md (the trigger and adjust-list insertion mechanisms), snippets.md (copy-and-paste settings for things not exposed in the UI), advanced.md (overriding behaviour via ptxprint-mods.tex), and ptx2pdf-faq.md (common stylesheet and macro questions). Where a TechRef entry has a fuller treatment elsewhere, that guide is the place to start; this manual remains the authoritative list of names, values, and defaults.

The PTXprint source may be found at <https://github.com/sillsdev/ptx2pdf>
This document is [docs/documentation/PTXprintTechRef.odt](#)

PTXprint Specific

Piclists

anchor caption|size="size" pgpos="pgpos" ref="ref" scale="scale" etc. % remainder

Anchor	Description
<i>bkk c.v</i>	e.g. JHN 3.16. Book chapter verse reference matching the \v value
<i>bkkL c.v</i>	As above but with diglot side reference
<i>bkk term</i>	A glossary item e.g. \k This Term\k* and GEN ThisTerm
<i>bkk id</i>	Milestone id as in \zfiga id="r12"* and GEN r12
<i>bkk bookstart</i>	Start of a book
<i>anchor=n</i>	Number of paragraphs after the anchor e.g. JHN 3.16=2

Size	Description
<i>full</i>	the entire size of the paper, a page reserved for images
<i>page</i>	the normal printed area of the page, on a page reserved for images
<i>col</i>	the width of the current column (only valid for 2 column text or diglot)
<i>span</i>	the width of the normal printed area of the page
<i>width</i>	the full width of the paper
<i>font</i>	the height of the font
<i>line</i>	the current line spacing

pgpos (regex)	Description
<i>t</i>	Top. Above everything except the header line.
<i>tr tl ti to</i>	Top of left or right or inner or outer column (if 2 columns)
<i>b[rlcio]?</i>	Bottom
<i>h[rlcio]?</i>	Here mid paragraph. right, left, inner, outer or *centre aligned
<i>p[rlcio]?</i>	After this paragraph right, left or *centre aligned
<i>p[rlcio]?#</i>	After the given number of paragraphs (here is paragraph 1)
<i>c[rlcio]?#</i>	Cutout left, right, inner or outer. Number is line count, may be fractional.
<i>P[lrcio]?[tbcf]?</i>	Page within margins with horizontal and vertical position (top, bottom, centre, fill - sidebars only)
<i>F[lrcio]?[tbcf]?</i>	Full page outside margins with horizontal and vertical position
<i>[tb][LR]</i>	Top or bottom of diglot column
<i>H</i>	HERE, character-like.
<i>B</i>	BELOW footnotes and anything else except footer.

Attribute	Description
<i>alt; caption;</i>	USFM mandated values: Alternative description or comment; Figure
<i>copy; ref</i>	caption; copyright information; reference for caption.
<i>scale</i>	Image linear scale factor with no scaling as 1.0.
<i>x-spacebeside</i>	Cutout image horizontal space to text including units. E.g. "3pt" Default is set by \DefaultSpaceBeside
<i>x-spacebefore</i>	Per image override of SpaceBefore in line units
<i>x-spaceafter</i>	Per image override of SpaceAfter in line units
<i>x-edgedjust</i>	Indent (or outdent) an image relative to the border
<i>x-xetex</i>	rotate=edge binding odd=degrees even=degrees; rotated degrees; page=sourcepdf_pnum
<i>media</i>	allowed media: a=applications, p=print, w=web
<i>mirror</i>	odd even both
<i>x-credit</i>	In image credit text

x-creditpos	[tcb][lcr]o]
x-creditrot	rotation of credit box. Default is bottom to edge
x-creditbox	<i>r g b</i> (0-1.0) <i>stylename</i> true t

Adjustment Lists

This section is much less relevant now that PTXprint creates and maintains the AdjList.

Historically, adjustments lists are held strictly in reference order. No other entries are matched until the next one in the list matches. A garbled line may cause crashes or following entries to be skipped. This behaviour saves memory and can be reverted to with the command \adjslurpfalse. The new behaviour loads all adjustments as the .USFM file is opened.

Syntax: *bkk C.V [+ -]n* also *bkk C.V [+ -]n[m]*, *bbk C.V [+ -]n[m f]*

where: *bkk* is a book code; *C* a chapter number; *V* a verse identifier following \v; + to increase the paragraph length; - to decrease the paragraph length (don't include the square brackets); *n* number of lines to increase/decrease; *m* paragraph number following reference, square brackets must be supplied around this number. The paragraph containing the reference is *m*=1. *f* references a note marker.instance with an optional .paragraph in the note. E.g. GEN 3.1 +1[ef.1 3] (First ef note in the first paragraph in GEN 3.1, the third paragraph in that note). Anything following % is ignored (including the %). Blank lines are then ignored.

Keyboard Shortcuts

Shortcut	Action
Ctrl-minus	Shrink - size of text in current view
Ctrl-equals	Grow + size of text in current view
Ctrl-S	Save file
Ctrl-D	Duplicate current line
Ctrl-L	deLete current Line
Ctrl-O	Comment out/in the current line (uses \rem, #, %)
Ctrl-,	Comment out/in the current line (uses \rem, %)
Ctrl-Up	Move the current line/selection UP by one line
Ctrl-Down	Move the current line/selection DOWN by one line
Ctrl-F	Find: Search for settings (does not work in the viewer/editor)
Ctrl-Q	Toggle Quick Run setting
Ctrl-R	Toggle cReate Diff setting (on Finishing tab)
Ctrl-P	Print (Make PDF)

Z-Codes

Marker	Description
\zbl[<i>num</i>]*	Blank lines
\zcopyright	Insert project copyright statement
\zgap[<i>dimen</i>]*	Insert a vertical gap of given dimension
\zimagecopyrights[<i>LANG</i>]	Insert image copyrights for a language
\zlicence	Insert projects license
\zcopyright	Insert project copyright statement
\zfiga <i>anchor</i> *	Insert image from piclist with given anchor

Marker	Description
<code>\zccimg by-nd size="size" pgpos="pgpos" scale="scale"*</code>	Insert Create Commons logo: by, by-nc, by-nc-nd, by-nc-sa, by-nd, by-sa; pgpos=p. E.g. <code>\zccimg by-nc-nd size="col" pgpos="pc" scale="0.2"*</code>
<code>\zpdf file="path/to/PDFfile" page="N" scale="NNNN" rotate="NNN"*</code>	Insert a PDF at the current location. file= param is mandatory. e.g. <code>\zpdf file=" ../../shared/orthographyStatement.pdf" page="3" scale="850" rotate="90"*</code> inserts just page 3, scale it to 85%, rotated by 90 degrees
<code>\zplaceallnotes</code>	Output end notes for this file here
<code>\zendnoterule</code>	Insert an endnote type rule, including spacing
<code>\zpostendnoterule</code>	If there were endnotes just now, insert a rule to mark the bottom.
<code>\ztoc[order]*</code>	Insert Table of Contents of given order (main, ot, nt, dc, pre, post, heb, sorta, sortb, sortc)
<code>\zvar variablename*</code>	Insert project variable value
<code>\zthumbtab bookname</code>	Set the thumb tab text for this book (instead of <code>\toc3</code>)
<code>\zlabel uniquelabel*</code>	Remember where <i>uniquelabel</i> is (page and reference)
<code>\zpage uniquelabel*</code>	Output the page number for <i>uniquelabel</i>
<code>\zref id="uniquelabel" show="b_c:v"*</code>	Output chapter:verse for <i>uniquelabel</i> (show: b=book, c=chapter, v=verse, _=space, anything else is copied)
<code>\zISBNbarcode isbn="num" height="size" var="varid" price="price" pricecode="code" pricecodevar="varid" pricevar="varid" font="Font Name" fontheight="12pt"*</code>	Insert ISBNbarcode for <i>num</i> (10 or 13 digits). Optional parameter size one of (normal, medium, short). If isbn is missing it tries the <i>zvar varid</i> . <i>Font</i> selects an alternative font; <i>fontheight</i> specifies the main (bottom) font height, to which everything is scaled. If <i>pricecode</i> is supplied then an extended ISBN barcode (EAN-5) is generated, falling back to <i>pricecodevar</i> , supplying pricing information required in some countries. The code is a 5 digit number, with the first digit representing the currency (0,1=GBP, 3=AUD, 4=NZD, 5=USD, 6=CanD) and the next 4 digits representing the price, according to local conventions. <i>price</i> is a human readable text string, which may give other information, e.g. an alternative currency, falling back to <i>pricevar</i> .
<code>\zendperiph</code>	End of a periph section, if you're not starting a new periphery. Must be the only thing on the line.
<code>\zgetperiph id*</code>	Get (use) the periphery stored earlier by <code>\periph comment id="intot"</code>
<code>\zglot[col]*</code>	Use formatting for diglot column <i>col</i> , in a monoglot section. Return to normal formatting with <code>\zglot *</code>
<code>\dopagenums</code>	Turn on page numbers
<code>\nopagenums</code>	Turn off page numbers
<code>\resetpagenums pnum</code>	Reset page numbers (negative for Roman numerals)
<code>\singlecolumn</code>	Required along with <code>\onebody</code> if forcing 1-col layout for section
<code>\onebody</code>	Set <code>\BodyColumns=1</code> for single column body text
<code>\doublecolumns</code>	Required along with <code>\twobody</code> if forcing 2-col layout for section
<code>\twobody</code>	Set <code>\BodyColumns=2</code> for double column body text
<code>\threecolumns</code>	Required along with <code>\threebody</code> if forcing 3-col layout
<code>\threebody</code>	Set <code>\BodyColumns=3</code> for three column body text
<code>\goone</code>	Switch to one column text now
<code>\gotwo</code>	Switch to double column text now
<code>\gothree</code>	Switch to three column text now
<code>\ztmon \ztmoff</code>	Turn on /off TeX's tracingmacros for debugging

Marker	Description
<code>\makeatletter</code>	Allow access to names with @ in them
<code>\makeatother</code>	Disable access to names with @ in them
<code>\m@kedigitsother</code>	Treat digits as digits
<code>\m@kedigitletters</code>	Treat digits as letters (for use in names)
<code>\beginL</code>	Start LTR text run
<code>\beginR</code>	Start RTL text run
<code>\fillsignature pages="8" pagenums="no" extra="2"*</code>	Fill a signature to the given modulo of pages. Note, uses the actual page count. Optional 'pagenums' parameter 'no'=\nopagenums, 'do'=\dopagenums), extra states that this many extra pages will be added manually or by special code before page lay-up.
<code>\ztocbefore ... \ztocbefore*</code>	Insert table rows (<code>\tr \cat toc\cat*... </code>) before the TOC
<code>\ztocafter \ztocafter*</code>	Insert table rows (<code>\tr \tc1 </code>) after the TOC.
<code>\zendtable</code>	Force a table to end and be processed, (as happens automatically when a new paragraph style is started). Can help before pagebreak, zgap, etc. when a normal paragraph break causes unwanted effects.
<code>\zEmptyPage</code>	Insert intentionally empty page contains \TPILB
<code>\zNeedOddPage</code>	Insert an "empty" page if the current page is even
<code>\zNeedEvenPage</code>	Insert an "empty" page if the current page is odd
<code>\zNeedQuadPage</code>	Insert 1, 2 or 3 "empty" pages if the current page is not divisible by 4
<code>\zifvarset var="varname"*</code>	Sets conditional text if <i>varname</i> is set. emptyOK="T" ⇒ empty is true, emptyOK="F" ⇒ empty is false.
<code>\zifhooks marker*</code>	Does <i>marker</i> have a hook, set conditional true
<code>\ztruetext ... \ztruetext*</code>	Include contents if condition is true
<code>\zfalsestext ... \zfalsestext*</code>	Include contents if condition is false
<code>\zcustommark type="user" value="text"*</code>	Inserts a mark of given type and value
<code>\zcustombotmark type="user"*</code>	Gets the value of the given mark at the bottom of this page. Used in headers/footers
<code>\zcustomfirstmark type="user"*</code>	Gets the value of the first given mark on this page. Used in headers/footers
<code>\zcustomtopmark type="user"*</code>	Gets the value of the given mark at the bottom of the previous page. Used in headers/footers
<code>\zrule cat="category" width="width" thick="thickness" align=" c r" raise="raise" style="mode"*</code>	Draws a line. <i>category</i> zrule category for style. Common list available. Missing defaults to simple line. <i>mode</i> =double for double line. <i>thickness</i> =height dimension. <i>width</i> =proportion of line. All attributes optional. <code>\zrule cat="VectorionRule1" width="0.4" align="c" thick="7pt"*</code>
<code>\zqrqcode cat="category" size="size" data="data" color="color" pgpos="pgpos" background="color" spacebeside="space" vmax="version"</code>	Insert a QR code of the <i>data</i> string. <i>size</i> =height dimension. <i>color</i> is foreground colour. <i>background</i> is background color. <i>spacebeside</i> is padding inside qrcode box. <i>pgpos</i> gives position on the page, see <i>piclist</i> parameters. <i>vmin</i> , <i>vmax</i> is QR code version limits, affecting size and error-correcting. Unicode specifies to use Unicode encoded text, which slows down processing (a temporary file must be written and re-read).

Marker	Description
<code>vmin="version"</code> <code>unicode="T F" *</code> <code>\zcolsync[id] *</code>	<code>\zqrcline size="1cm" pgpos="cr" data="https://unicode.org" *</code> Synchronisation point for diglots. If id is v2, then that point will be considered the start of verse 2. If id is p2 then that point will be treated as the start of the second paragraph, even if it's actually the 4 th . Alternatively matching entries (which must occur in ASCII order) can be given in each file.
<code>\zsetreflbook="book"</code> <code>bkid="GEN"</code> <code>chapter="1"</code> <code>verse="3" *</code> <code>\zexp expansion</code> <code>\SHOWSTUFF</code>	Set the running header to <i>book</i> , the book id to <i>GEN</i> , the chapter to 1, and verse to 3. This does not itself cause any triggers, but it means that the next verse will fire triggers for GEN 1:whatever, rather than MOD 0:whatever from a module. Used in triggers to set the text font expansion in percent, e.g. 96 Produces no output in PDF, but write most of the significant flags, etc to the logfile. A useful debugging aid.

Table of Contents types

A type may be followed by a diglot side identifier as in sortcR

Type	Description
main	Default page number order of all books [default]
ot	Old Testament
nt	New Testament
dc	Deuterocanonical books
pre	Before Old Testament books (e.g. FRT)
post	Post scriptural books (e.g. GLO)
heb	Old Testament books in Hebrew order
sorta	Sort by \toc1
sortb	Sort by \toc2
sortc	Sort by \toc3

Marker Style Fields (.sty Files)

Field	Description
\Marker	The start of a marker description
\Endmarker	Specifies the closing marker for this group
\Name	3 parts separated by -: marker, category, summary description
\Description	Textual one line description of the marker
\OccursUnder	List of markers this marker may occur under in the hierarchy
\TextProperties	(non)publishable, vernacular
\TextType	VerseText, NoteText, BodyText, Title, Section, Other, ChapterNumber, VerseNumber
\StyleType	Paragraph, Character, Note, Milestone
\[Mm]rkttype	USFM 3.1 marker category
\Color	decimal B * 65536 + G * 256 + R, or xRRGGBB hex digits
\FontName	font name to pass to font specification
\FontSize	actual font size is \FontSize * \FontSizeUnit. 12 = main font size
\FontScale	Scaled font size relative to the current font (e.g. \p)
\texFontFeatures	XeTeX style font feature settings to append to the font name specification when a font is instantiated
\ztexGrSpace	The \XeTeXinterwordspacing value to use when this marker is active
\Italic	blank enables, “-” disables
\Bold	blank enables, “-” disables
\Superscript	blank enables, “-” disables. Equivalent to \FontScale \SuperscriptFactor, \Raise \SuperscriptRaise
\SmallCaps	blank enables, “-” disables
\Regular	Disables \Italic, \Bold, \Superscript
\Raise	Specifies a dimension to raise the text by. Be careful, this can't handle whole paragraphs and will not allow line breaking within the text so marked
\BaseLine	(see also LineSpacing.) Dimension of line spacing. An absolute measure that can include glue. (e.g. `12pt plus 1pt minus 1pt'). <i>Units are required</i>
\LineSpacing	Dimension of the line spacing, as a proportion of the default line spacing (1.0 is one line). i.e. \LineSpacing * \LineSpaceBase * \LineSpacingFactor * \FontSizeUnit
\FirstLineIndent	First line indent * \IndentUnit. 1 = Specified indent factor. For sidebars, this indents (or outdents if negative) a the sidebar.
\LeftMargin	Left margin * \IndentUnit (Right margin in RTL)
\RightMargin	Right margin * \IndentUnit (Left margin in RTL)
\Justification	values; center, left, right. Anything else is fully justified
\NonJustifiedFill	Proportion of line to allow for line break filling. Default 0.25
\SpaceBefore	Space before paragraph (\SpaceBefore * \VerticalSpaceFactor * \LineSpacingFactor * \FontSizeUnit)
\SpaceAfter	Space before paragraph (\SpaceAfter * \VerticalSpaceFactor * \LineSpacingFactor * \FontSizeUnit)
\CallerStyle	Marker to style the body text caller for a note
\CallerRaise	Dimension to raise the body text caller
\NoteCallerStyle	Marker to style the in note caller
\NoteCallerRaise	Dimension to raise the in note caller

Field	Description
<code>\NoteBlendInto</code>	name of note type marker to merge these notes into that class, but with styling specified here
<code>\Underline</code>	1 = single underline. 2 = double underline ¹
<code>\Background</code>	Apply a background colour or highlight ¹
<code>\Outline</code>	Line thickness (* FontsizeUnit) to outline text with ¹
<code>\OutlineColour</code>	Colour of outline line ¹
<code>\Shadow</code>	Offset of text shadow multiplied by font size ¹
<code>\ShadowColour</code>	Colour of text shadow ¹

Sidebar/Box Style Fields

Field	Description
<code>\Position</code>	Position on the page: [tbhpfB][lricio]? [FP]([lricio][ctbf])? default bl
<code>\Scale</code>	Proportion of width of parent
<code>\SpaceBeside</code>	Horizontally shift the sidebar relative to the margin, in same units as paragraph indentation. May be negative.
<code>\SidebarGridding</code>	normal – top of sidebar is top of a subpage, last line is on grid; smart – as normal for sidebar of integral number of lines; orig – obey spaceBefore; *none – no attempt at gridding, ignore spaceBefore; heading – as none but obey spaceBefore
<code>\BgColor</code>	F (disable), T (enable), or 3 values 0-1.0 RGB
<code>\Alpha</code>	Background transparency
<code>\FgImage</code>	filename of image (.jpg, .pdf)
<code>\FgImagePos</code>	Location of image within box
<code>\FgImageScale</code>	How much to scale image by (default 0.2)
<code>\BgImage</code>	filename for horizontally and centred background image
<code>\BgImageScale</code>	floatxfloat or float or xfloat for image scaling
<code>\BgImageOversize</code>	ignore (overflow, no crop), (shrink, distort, crop)
<code>\BgImageLow</code>	t (under background color) or f (over background color)
<code>\BgImageAlpha</code>	Alpha for background image
<code>\BgImageColor</code>	3 floats (RGB) or x with 6 hex digits
<code>\BgImageCropOfs</code>	Proportion of image to shift when cropping
<code>\BoxPadding</code>	margin between border and background color/image
<code>\BoxPaddingInnerOuter</code>	T – Left means inner, Right outer; F – Left and Right are left and right.
<code>\BoxHPadding</code>	Horizontal margin
<code>\BoxVPadding</code>	Vertical margin
<code>\Border</code>	Apply border to this side: None, Top, Bottom, Left, Right, Inner, Outer, All
<code>\BorderWidth</code>	Total border width including 2 lines and fill, etc. In pts
<code>\BorderColour</code>	3 floats (RGB) or x with 6 hex digits
<code>\BorderStyle</code>	plain, double, ornaments. Describes how many lines are in the border. May also reference a standard style, e.g. Vectorian1, in which case it behaves as <code>\BorderRef</code>

¹ Should only be applied to character styles and prevents line breaking within the character style. Paragraph style support is not yet implemented

Field	Description
<code>\BorderRef</code>	Reference (inherit values from) another (e.g. standard) style. Parameters not defined for the current style will be used from the named style. Resolution of inherited values occurs on reading the first USFM file.
<code>\BorderFillColor</code>	3 floats (RGB) or x with 6 hex digits. Color to fill between border lines or in a complex ornament
<code>\BorderLineWidth</code>	Thickness of lines in the border in pts
<code>\BorderPadding</code>	External padding margin to the border (from outside text, say). In pts.
<code>\BorderHPadding</code>	Horizontal external padding
<code>\BorderVPadding</code>	Vertical external padding
<code>\BorderPatternTop</code>	Border pattern syntax description of top border of box
<code>\BorderPatternBot</code>	Border pattern syntax for bottom border of box
<code>\BorderPatternLeft</code>	Border pattern syntax for left border of box
<code>\BorderPatternRight</code>	Border pattern syntax for right border of box
	The left and right borders match the height of the contents (plus padding), and the top and bottom patterns provide the corner pieces. The sequence of creation (if they are to be used) is Top, Bottom, Left, Right
<code>\OrnamentScaleRef</code>	Define (comma separated) named scale factors for this border based on the natural size of the given ornament(s). Format is <i>name:type reference_orn[:type reference_orn]</i> <i>type</i> is one of x, y, xy, X,Y,XY. Upper case sets the scale based on the other dimension of the reference ornament. e.g. <i>corner:X/33:y/32</i> defines the named scaling <i>corner</i> to be such that the natural height of orn. 33 defines the x scaling and the natural height of orn.32 defines the y scaling.

Stylesheet Processing Order

Later values override earlier ones

File

<code>usfm_sb.sty</code>	Purpose Base USFM specification stylesheet
<code>prj/usfm.sty (*)</code>	Project specific extensions to USFM
<code>ptx2pdf.sty</code>	PTXprint USFM stylesheet extensions
<code>prj/custom.sty (*)</code>	Project custom stylesheet
<code>shared/ptxprint/prj/cfg/ptxprint.sty (*)</code>	Project stylesheet from PTXprint UI
<code>shared/ptxprint/prj/cfg/ptxprint-mods.sty (*)</code>	Project stylesheet overrides
* processes all the corresponding files for each glot	

Border pattern syntax

- Pattern element is: number or "string"|rotation(uvhdrlR)|fill(x-) or repeat(?+(i,j)"=)[A-Za-z]]scaling(float. or name)|adjustment. Any final empty | may be ignored. **u** is unmodified
u - up-> **up**, d - up->**down**, l - up->**left**, r - up->**right**.
h - Mirror left->right, v - Mirror up-down
L and R apply rotation and then **h** mirroring.
- Ornament string is , separated list of pattern elements

u:  l:  r:  d: 
h:  L:  R:  v: 

Marker Grammar

Example marker: id:XXS|periph:strongshebrew|cat:emphasised|k+s

marker: markerexp ('+' (tag | ms))*

markerexp: (constraints '|')? tag | ms

constraints: categories | periph | id

categories: category (':' category)*

category: 'cat:' catval

periph: 'periph:' catval ('|' categories)?

id: 'id:' catval ('|' (periph | categories))?

ms: (defaultval '|')? tag

tag: idchar+

catval: idchar+

defaultval: uniwdchar*

idchar: [a-zA-Z0-9_] | '-'

uniwdchar : [^\s"]{+}

Milestones are treated as character styles and propagate into paragraphs that occur while active.

If not clear from the above constraints (id, periph: and cat:) must occur in that order. If the tag1+tag2+tag3 format is used, tag3 is the outermost (e.g. a paragraph) and tag1 the innermost. E.g. to style \p ... \wj ... \+nd the style identifier is nd+wj+p

Key Styles

Style Name	Where	Description
zglm	S	Glossary marking characters
xts	S	Strong's digit cell
cat:ornaments zrule	S	Border styling for \zrule cat="ornaments"
cat:frontpage esb	S	Sidebar styling for \esb\cat frontpage\cat*
cat:toc tr	S	TOC table row styling
cat:toc tc1	S	Column 1 cell styling for TOC (also for tc2, tc3, etc.) not tcr3
cat:foo tr	S	Styling for \tr\cat foo\cat* and subsequent rows in the same table.
wg+strong-s	S	Strong's Greek term (also k, wh, wl, xt) (\wg is a char style)
strong-s+li	S	List entry paragraph styling for Strong's Index (also m) (\li is a paragraph style)
periph:cover mt1	S	Styles \mt1 for the cover
id:MAT ip	S	Styles \ip in the book of Matthew
textborder	S	If enabled, styling for automatic border around (verse-containing) body text.
cat:headingsbox esb	S	If enabled, automatic sidebar which encloses all heading blocks
cat:titlebox esb	S	If enabled, automatic sidebar which encloses all title blocks
\p^95	U	^ prefix for font expand value in %. E.g. 0.95 expand = 5% compression. Also p^103 to widen text.

S: automatically applied style, use the name in style definition.

U: Automatically interpreted style, use the name in USFM for the special effect.

Headers

Without odd or even, a header or footer is the fallback if either the odd or even is undefined.

Header/Footer	Usage
<code>\R(H F)(odd even)?(left right center)</code>	Non first page, containing scripture
<code>\R(H F)title(odd even)?(left right center)</code>	First page of a book (ptxfile)
<code>\R(H F)noV(odd even)?(left right center)</code>	Non first page, not containing scripture

Macro	Description
<code>\pagenumber</code>	The page number. Maybe disabled with <code>\nopagenums</code>
<code>\folio</code>	The page number. May not be disabled.
<code>\rangeref</code>	First and last references on the page
<code>\firstref</code>	First reference on the page
<code>\lastref</code>	Last reference on the page
<code>\book</code>	Book name from <code>\h</code>
<code>\bookalt</code>	Alternate book name from <code>\h1</code>
<code>\usdate</code>	Date in month/day/year format
<code>\ukdate</code>	Date in day/month/year format
<code>\isodate</code>	Date in year-month (2digits)-day format
<code>\year</code>	Year
<code>\month</code>	Month number
<code>\day</code>	Day number
<code>\ROMANnumeral#1</code>	Converts the parameter to uppercase roman numerals
<code>\romannumeral#1</code>	Converts the parameter to lowercase roman numerals
<code>\timestamp</code>	Date and time as year.month.day :: hours:minutes (each, except year, 2 digits). As used in crop marks.

Changes syntax elements

Id	Element	Description
replace	<code>"regex" > "output"</code>	Simple replacement
replace	<code>'regex' > 'output'</code>	Simple replacement
inreplace	<code>in 'regex' inreplace replace</code>	Matches regex and applies replacement within regex
atreplace	<code>at reflat inreplace replace</code>	Within the USFM covered by the references in the reflat, apply the replacement

References

A reference list has this grammar:

Id	Expression	Description
verse	<code>\d+[a-z]?</code>	Verse number with possible sub verse
chapter	<code>\d+</code>	Chapter number
CV	<code>chapter([.:]verse)?</code>	Chapter with verse
book	<code>\S{3}</code>	Book id from list of books, e.g. JHN
ref	<code>book CV CV verse</code>	A simple complete reference
range	<code>ref-ref</code>	A range of two references
reflist	<code>(range[.,;]\s*)+</code>	List of references

e.g. JHN3:16-20; LUK 1, 2:15, 19; JON 2:3-4:6, 1JN 1:9

Other changes file content

include "path/to/changesfile.txt" Include the named file

sections(**step**[, **step**])

The changes file can now be divided into distinct sections. Those sections apply at different points in the processing flow (many of the events are optional):

Changes step	Preceding event(s)	File
periph default	File is read	FRT, INT
initial	Processing with a "before" script. USFM file is read.	normal
default	Chapter range check, Interlinear reconstruction, Module flattening	normal
final	Canonicalise USFM, Introduction removal, Empty verse removal, End of ayah, Strong's numbers, Manual hyphenation, Other UI controlled changes, Apply paragraph stretch.	normal
merged	Merging files into a typesetting single file	diglot
piclist	the locally generated piclist	piclist
module	module is read before flattening	module

Peripherals

If stored, the macros hold onto the peripheral contents to output it in the right place or in response to \zgetperiph. Use \NoStorePeriph{id} or \StorePeriph{id} to change behaviour.

Id	Stored	Book	Description
abbreviations	no	FRT	Abbreviations used and their description
alphacontents	no	FRT	Alphabetical Contents section, if separate from contents
intbible	yes	INT	Introduction to the Bible
chron	no	BAK	Chronology
contents	no	FRT	Contents page(s)
cover	no	FRT	Cover page. <i>PTXprint</i> uses coverfront
coverback	yes	FRT	Back page of the cover
coverfront	yes	FRT	Front page of the cover
coverspine	yes	FRT	Spine of the cover
coverwhole	yes	FRT	The structure of the whole cover
foreword	no	FRT	The foreword section
halftitle	no	FRT	The half title page between the title page and the publication data
imprimatur	no	FRT	For an imprimatur
intbible	yes	INT	Introductory section to the Bible
intdc	yes	INT	Introductory section of the Deuterocanonical books
intepistles	yes	INT	Introductory section to the Epistles (before ROM)
intgospels	yes	INT	Introductory section to the Gospels
intletters	yes	INT	Introductory section of the letters
intnt	yes	INT	Introductory section to the New Testament
intot	yes	INT	Introductory section to the Old Testament
intpent	yes	INT	Introductory section to the Pentateuch
intpoetry	yes	INT	Introductory section to the the Poetry books
intprophecy	yes	INT	Introductory section to the Prophetic books
insert_id	yes	INT	Inserts for extraction to other print files
lxxquotes	no	BAK	LXX quotes in NT (back matter)

maps	no	BAK	Map Index
measures	no	BAK	Weights and Measures
preface	no	FRT	Preface
promo	no	FRT	Promotional text, e.g. for back cover. PTXprint uses coverback
pubdata	no	FRT	Publication data including copyrights, etc.
spine	no	FRT	Spine contents. PTXprint uses coverspine
title	no	FRT	Title page (non cover)

Page numbering and numbering resets

By default, page numbering in peripheral books uses roman numerals (i, ii, ...) until the intbible / intpent/intnt peripherals. Internally, roman numerals are selected by using a negative page number. Before the intbible periph is read, blank pages are inserted to ensure that the they start on an odd numbered page, and if the numbering is negative, numbering restarts at 1 (or the value selected by the configuration). If (due to user intervention) the numbering is negative just before the first scripture book (e.g. GEN), then the page number reset is triggered again.

(Both of) these resets preserve the previous page number in the \oldpageno macro. Thus support for alternative page numbering systems is provided.

For instance if it is desired that the reset happens at the beginning of the first scripture book, the user can undo the reset at intbible, or those preferring a layout which does not have any resets at all (just starting with 1) can start the count at 1 from the start of the periphery.

If it is desired to not have a blank page before intbible (etc), then the peripherals can be pulled in from the FRT book.

\pageno=\oldpageno	Undo a page number reset (put this at the start of the intbible (or intnt, etc) periphery book).
\resetpagenums 1	Restart the count at one.

Command-line Options

usage: **ptxprint** [-h] [-b BOOKS] [-c CONFIG] [-R RUNS] [-P] [-p PROJECTS] [-d DIRECTORY] [-L LANG] [-Z ZIP] [-A ACTION] [-o OUTPUT] [-m MACROS] [-M MODULE] [-S] [-T] [-C CAPTURE] [-t TEST] [--testwithgui] [-j JOBS] [-q] [-l LOGGING] [--logfile LOGFILE] [-to TIMEOUT] [--debug] [-f FONTPATH] [--nofontcache] [--testsuite] [-F DIFFFILE] [--diffpages DIFFPAGES] [--diffoutfile DIFFOUTFILE] [--diffdpi DIFFDPI] [-V PDFVERSION] [-N] [-n PORT] [--nox11] [-D DEFINE] [-z EXTRAS] [-i] [-E EXPERIMENTAL] [pid] [{print,fill,archive}]

positional arguments:

pid	Project id or full path to a ptxprint.cfg file
{print,fill,archive}	Non interactive command to execute

Option

-h, --help
 -b BOOKS, --books BOOKS
 -c CONFIG, --config CONFIG

Description

show this help message and exit
 Reference list to print (as in Choose Books)
 Configuration path to load

-R RUNS, --runs RUNS
 -P, --print
 -p PROJECTS, --projects PROJECTS
 -d DIRECTORY, --directory DIRECTORY
 -L LANG, --lang LANG
 -Z ZIP, --zip ZIP
 -A ACTION, --action ACTION
 -o OUTPUT, --output OUTPUT
 -m MACROS, --macros MACROS

 -M MODULE, --module MODULE
 -S, --enablescripts
 -T, --testing
 -C, --capture CAPTURE
 -t, --test TEST
 --testwithgui
 -j JOBS, --jobs JOBS
 -q, --quiet
 -l LOGGING, --logging LOGGING

 --logfile LOGFILE
 -to TIMEOUT, --timeout TIMEOUT
 --debug
 -f FONTPATH, --fontpath FONTPATH

 --nofontcache
 --testsuite
 -F DIFFFILE, --difffile DIFFFILE
 --diffpages DIFFPAGES
 --diffoutfile DIFFOUTFILE
 --diffdpi DIFFDPI
 -V PDFVERSION, --pdfversion PDFVERSION
 -N, --nointernet
 -n PORT, --port PORT
 --nox11
 -D DEFINE, --define DEFINE
 -z EXTRAS, --extras EXTRAS

 -I, --identify
 -E, --experimental EXPERIMENTAL

limit xetex runs
 Hits print
 Path(s) to Projects directories (repeatable)
 Directory to store temporary files
 Set UI language code
 Unzip into project directory and delete at end
 Run view method instead of print
 Output file for actions like archive
 Directory containing TeX macros (paratext2.tex)
 Specify module to print
 Enable process scripts in CLI use
 Run in testing, output xdv
 Capture interaction events
 test file to run interactive test against
 Run test with visible gui and don't exit
 Number of multiprocessing jobs to run
 Suppress splash screen and limit output
 Logging level [DEBUG, *INFO*, WARN, ERROR, number]
 Set logging file [ptxprint.log] or "none"
 XeTeX runtime timeout (seconds)
 Enable debug output
 Specify directories containing fonts (repeatable)
 Don't use system fonts
 Use only fonts from project and test suite
 Generate difference PDF against another file
 Max pages to insert in diff file
 Output filename for difference PDF
 DPI for PDF differencing
 PDF Version to read/write (default=14)
 Disable all internet access
 Listen on given port
 Don't switch to X11
 set UI component=value (repeatable)
 Special flags (verbosity of xdvipdfmx, request PTdir, no config)
 add widget and config names to tooltips
 Enable experimental features (0 = UI extensions)

Logging

Where the commandline is not accessible, logging can be enabled by creating a file **ptxprint_logging.cfg** in C:\Users\<Username>\AppData\Local\SIL\ptxprint on Windows or ~/.config/ptxprint on Linux. An example file is:

```
[loggers]
keys=root

[handlers]
keys=hand01

[formatters]
keys=form01

[logger_root]
level=DEBUG
handlers=hand01

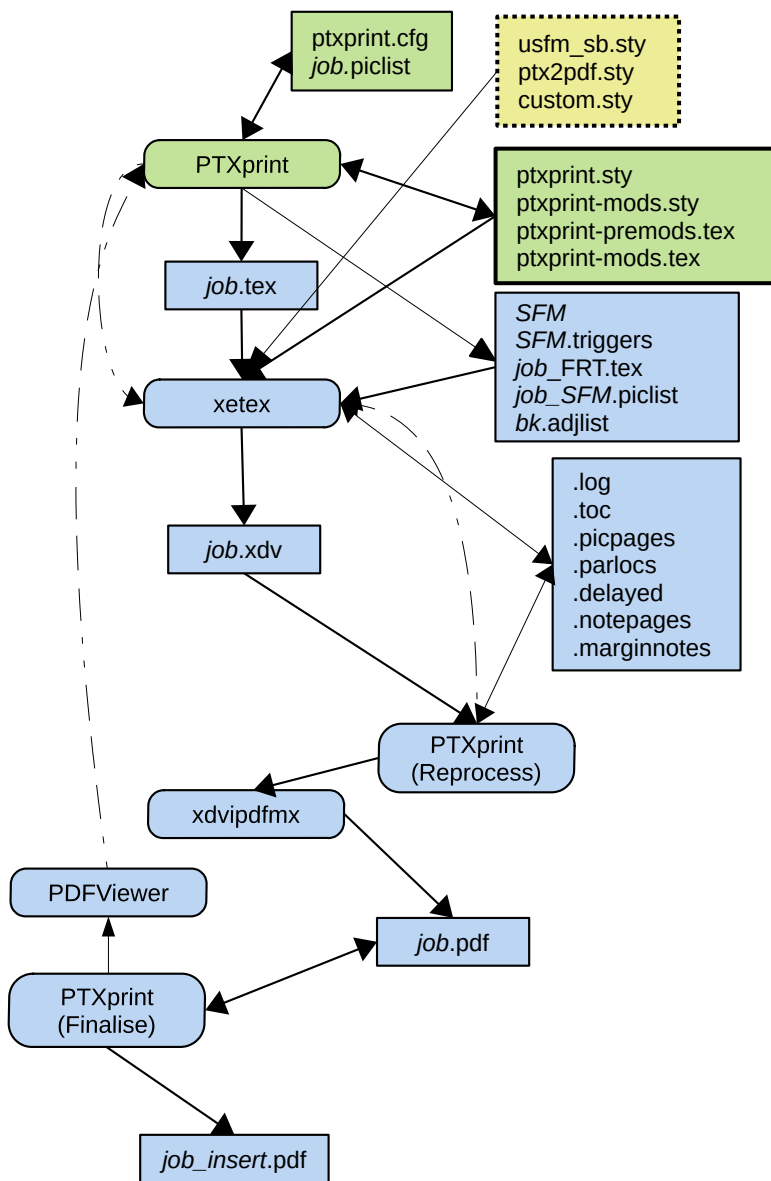
[handler_hand01]
class=FileHandler
level=DEBUG
args=('c:/<writeablepath>/ptxprint_debug.log', 'w')
formatter=form01

[formatter_form01]
format=%(asctime)s.%(msecs)03d %(levelname)s: %(module)s %(lineno)d %(message)s
datefmt=%d/%b/%Y %H:%M:%S
class=logging.Formatter
```

Diglot merge debugging

If the debug level is 5 or less, diglot merge verification files are created, one for each glot with each chunk in it. For example, if *19PSAprimary.SFM* and *19PSAsecondary.SFM* are the (processed) primary and secondary input files, then the merged (final SFM) file is *19PSAprimary-diglot.SFM*, and the verification files, *19PSAprimary-diglot.SFM-L* and *...-R*. These can then be diffed against the originals, e.g.

```
wdiff -3 19PSAsecondary.SFM 19PSAprimary-diglot.SFM-R
```



File Processing and Generated Files

[r] indicates the file is also read

Extension	Description
.tex [r]	Job file that XeTeX runs
.log	Generated log of the last XeTeX run against the .tex file
.delayed [r]	Paragraph positions of cutouts (drop chap, figures, sidebars). Is used to calculate paragraph cutouts: $\backslash\text{DelayedItem}\{type\}\{ref\}\{pageno\}\{(\text{width}\times\text{length in lines}@delay)side\}$ $\backslash\text{RaiseItem}\{type\}\{ref\}\{pageno\}$
.parlocs [r]	Position of the start and end of each paragraph on the page $\backslash\text{@pgstart}\{pageno\}\{available\ height\}\{pagewidth\}\{pageheight\}$ $\backslash\text{@parpageend}\{bottomx\}\{bottomy\}\{type=[bottomins, notes, verybottomins, pageend]\}$ $\backslash\text{@colstart}\{height\}\{depth\}\{width\}\{topxpos\}\{topypos\}$ $\backslash\text{@parstart}\{ref\}\{type\}\{mkr\}\{baselineskip\}\{xpos\}\{ypos\}$ $\backslash\text{@parend}\{badness\}\{bottomx\}\{bottomy\}\{lastdepth\}$ $\backslash\text{@parlen}\{ref\}\{parnum\}\{number\ of\ lines\ in\ paragraph\}\{mrk\}\{adjustment\}$ $\backslash\text{@colstop}\{xpos\}\{ypos\}$ $\backslash\text{@Poly@colstart}\{height\}\{depth\}\{width\}\{topx\}\{topy\}\{polycode\}$ $\backslash\text{@Poly@colstop}\{bottomx\}\{bottomy\}\{polycode\}$ $\backslash\text{@parpicstart}\{ref\}\{src\}\{x\}\{y\}$ $\backslash\text{@parpicstop}\{ref\}\{src\}\{width\}\{height\}\{x\}\{y\}$ $\backslash\text{@parpicsize}\{width height\}\{heightlimit?\}\{width\}\{height\}$ $\backslash\text{@parpicanchor}\{ref\}\{picid\}\{x\}\{y\}$ $\backslash\text{@parnote}\{type\}\{caller\}\{ref\}\{index\}\{callerx\}\{callery\}$ $\backslash\text{@notebox}\{type\}\{width\}\{height\}$ $\backslash\text{@parlines}\{numlines\}$ $\backslash\text{@nontextstart}\{x\}\{y\}$ $\backslash\text{@nontextstop}\{x\}\{y\}$
.picpages [r]	Tracks which figures are on which pages to catch them moving $\backslash\text{figonpage}\{page\ number\}\{figure\ number\}\{file\ name\}\{pgpos\}\{copyright\}$ $\{misc:size, rotation, etc.\}$
.notepages [r]	For per page callers, holds the first caller per page: $\backslash\text{noteonpage}\{caller\}\{pageno\}$
.marginnotes [r]	Position of each margin note on the page $\backslash\text{@marginnote}\{ref\}\{marker\}\{side\}\{distance\ from\ text\}\{width\}\{height\}\{xshift\}$ $\{yshift\}\{pageno\}\{xpos\}\{ypos\}$
.fontusage	List of fonts used if $\backslash\text{ifLogFonts}$ true
.toc [r]	Table of Contents after processing to give different lists
_orig.toc	Copy of table of contents as generated from TeX
_FRT.sfm [r]	Generated front matter to be output
_INT.sfm [r]	Generated INT file of peripheral matter
.xdv	TeX device independent output. Generates PDF
.xdvi_log	Logs output from xdvipdfm-x, xdv to PDF conversion

Content-defining macros

Macro	Purpose
$\backslash\text{ChapterVerseSeparator}$	Separator between chapter and verse in header.

Macro

`\BookChapSeparator`
`\RangeSeparator`
`\RangeChapSeparator`
`\MakeChapterLabel#1#2{#1\ #2}`
`\printchapter#1{#1}`
`\printaltchapter#1{(#1)}`
`\AdornVerseNumber#1{#1}`
`\DecorateRef#1{(#1)}`
`\ISBN#1`
`\DefineActiveChar#1#2`
`\bridgeVerses bkkC.V-V.`
`\EndNoteSeparator`
`\zBookEndDecoration`
`\zautoBookEndDecoration`

Purpose

Separator between book and chapter in header.
 A range of verses in header
 If a range includes a change of chapters/books in header
 Sequence and separator for chapter labels.
 Any ornamentation for chapter numbers
 Any ornamentation for alternative chapter numbers
 Any ornamentation for verses
 Any decoration of the reference in a figure caption
 Output ISBN from TeX. (see also `zISBNbarcode`)
 Mark a character as being active with the given definition
 Bridges a pair of verse. Note the final . delimiter
 Makes the rule used before (and after) end-notes
 The user-interface-selected decoration at end of a book.
 Template automatically includes this at end of each book.
 When a user-selected decoration is included it is:
`\iffilehasverses\zBookEndDecoration\fi`
 Macros (default empty) to draw a rule or include other
 content after the main content of the book, assuming the
 last page was not so full that it has been written to the
 PDF file already by the transition to single-column.
`\lastbookendrule` is used after the final USFM file,
 otherwise `\bookendrule` is used. This content goes before
 any book-end hooks and automatic end-notes, which in
 turn go before `BookEndDecoration`

Triggers

Trigger files are included from `shared/ptxprint/config/triggers` (unless `\TrigListPath` has been changed), and/or the job directory. The filename consists of the job file or the filename passed to `\ptxfile` with `.triggers` appended. After a `\ptxfile` file has been opened successfully, `filename-1.triggers` is tried with the number increasing for each successful read.

Triggers are normally run once and then forgotten. Each takes a name and USFM content:

`\AddTrigger name`
`USFM content`
`\EndTrigger`

Pattern

`REV20.14-preverse`
`GLOk.LakeoffFire-preverse`
`ms:zif`
`ms:zwombat=a542`
`GENa542`
`REV20.14`
`GLOk.Herod`
`name=2`

Description

Triggers before the verse number
 Triggers before a glossary entry in the the GLO book
 Triggers before any standalone milestone of type `zif`: `\zif`
`file="wombats"*` in any book
 Triggers before the unique standalone milestone `\zwombat`
`id="a542"*`
 Triggers before any standalone milestone with matching id e.g.
`\zwombat|a542|*` in the book GEN
 Triggers mid-paragraph, after the post-versenumber space.
 Triggers mid-paragraph, after the glossary entry `\k Herod\k*`
 Triggers before the second paragraph after `name`

Global Settings and Commands

Global (layout-altering and *debugging*) booleans

For \ifanif, use \ifanif to test, \aniftrue to set and \aniffalse to unset.

Setting	Purpose
\ifActions	Make links active inside PDF [true]
\ifAlwaysCheckAttributes	Check all char styles for link-href attributes
\ifAttrMilestoneMatchesUnat tr	Should styling specified for a milestone without an attribute be applied to a milestones with an attribute? If true, then styling specified for an e.g. \qt-s* is also applied to \qt-s Pilate*. [false]
\ifBindingGutter	Is there a binding gutter [false]
\ifBookOpenLeft	Page order is reversed for RTL type books [false]
\ifbookresetcallers	Reset note caller sequences at the start of a book
\ifCalcChapSize	Attempt to automatically calculate drop chapter number size [true]
\ifCaptionFirst	Output caption before image [false]
\ifCaptionRefFirst	Output reference before caption [false]
\ifColNotes	Whether to have column notes [false]
\ifColNotesRule	Output a rule separating column notes from body text [false]
\ifColorFonts	Allow coloured fonts [true]
\ifColumnGutterRule	Output a central column gutter rule [false]
\ifCropMarks	Whether to insert crop marks (and increase page size) [false]
\ifdiglotMergedNotesRTL	Whether merged footnotes should be RTL
\ifDoCaptions	Include captions in figures
\ifDropActions	Don't output PDF bookmarks [false]
\ifendbooknoeject	Don't insert a page break after this book
\ifEndNotesSingleCol	Output endnotes in single column text
\iffigloleft	Default figure positions to top left [true]
\ifFigurePlaceholder	Insert place holders instead of picture files [false]
\ifFinalNotesDown	Push notes on the final page to the bottom of the page [false]
\ifHangeVA	Also hang alternative verse numbers
\ifIncludeFigures	Include figures at all [true]
\ifIndentAfterHeading	Allow indented paragraphs following a heading [true]
\ifIndentAtChapter	Allow indented paragraphs at chapter start with cutouts [false]
\ifJoinGutterRule	Join diglot text rule with notes rule [false]
\ifJustifyPars	Justify text [true]
\ifkeepmargininversemarkers	Insert in text markers for margin verses not output.
\ifLogFonts	Generate an extra .fontusage file showing which page each new character style is first used on and what font it uses.
\ifMarkAdjustPoints	Display an adjlist id for each paragraph [false]
\ifMarkTriggerPoints	Display trigger points in output [false]
\ifMidPageFootnotes	Should footnotes go before a single-double column transition [false]
\ifNoHangVerseNumberOne	Don't hang verse 1 if output with a drop chap
\ifnoinkinmargin	Reduce text block to insert footer rather than in the margin [false]
\ifnoparstartmvm	Whether to exclude verse 1 in marginal verses
\ifNoteGutterRule	Insert a rule between diglot columns [true]

Setting

`\ifnotesEachBook`
`\ifNoTransparency`
`\ifnotocwrite`

`\ifOmitBookRef`
`\ifOmitChapterNumber`
`\ifOmitChapterNumberRH`
`\ifOmitVerseNumberOne`
`\ifparnoteskillprevdepth`
`\ifparnotesmidtopskip`
`\ifparnotesruletopskip`
`\ifrefbookmarks`
`\ifsquashgridbox`

`\ifStudyGutterRule`

`\iftildenbsp`

`\ifstretchtab`

`\ifTOCthreetab`

`\ifTabsOddOnly`

`\ifUnderlineSpaces`

`\ifversehyphen`

`\ifVerseRefs`

`\ifVisTrace`

`\ifVistTraceExtra`

`\ifXrefSideAlign`

`\ifXrefTopFill`

Purpose

Output endnotes at the end of each book [true]

Disable transparency output in PDF [false]

Stops the macros writing a new .orig_toc file. Good for TOC design testing.

Omit the book name in the running header reference [false]

Don't output chapter numbers [false]

Omit chapter numbers from running header [false]

Don't output verse 1 at the start of a chapter [false]

Sets final note line baseline on the bottom of the note box

gridfits the top of the mid note skip

gridfits the note rule above the note

Use book codes instead of book names in bookmarks [false]

Don't insert space above headings at the start of a column [true]

Output a rule separating study notes columns [false]

Treat ~ as non breaking space [true]

Do tabs stretch if the text is too large? [false]

Use \toc3 for tab text if no \zthumbtab [true]

Only print thumb tabs on odd pages to ease ink seepage.

Underline spaces in underlined runs [true]

In marginal verses, do we insert a hyphen between verse ranges? [true]

Include verses in references [true]

Insert visible trace marks in diglot output [false]

Add extra information to diglot trace marks (cat alter layout) [false]

Align column notes to side and page [false]

Sit column notes on the bottom of the page

Settings in ornaments plugin

`\ifboxorn`

Display boxes around ornaments [false]

`\ifClipOrnaments`

Ornaments by default clip themselves to a certain box. That box is sometimes just too small for real beauty, so clipping has been disabled. `\ClipOrnamentstrue` restores previous behaviour.

`\ifCacheOrnaments`

Use (lots of) TeX boxes to store objects to reduce repeated loading [false]

`\ifXformOrnaments`

Use PDF's xform objects to reduce PDF size and repeated loading (experimental) [false]

Global Dimensions and Counts

For `\ADimension`, use `\ADimension=20pt` (any dimension) or `\ACount=number`

Dimension

Default

Description

`\AboveEndNoteSpace`

14pt

Space between text and end notes

`\AutoCallerNumChars`

26

Length of autocaller fallback sequence

`\AutoCallerStartChar`

97

Start of auto caller fallback sequence

`\badspacepenalty`

100

Penalty for a `\BADBREAK`

`\baselineskip`

Current line spacing

Dimension	Default	Description
<code>\lineskiplimit</code>	0.9 x lineheight	Allowed line overlap before forced spacing
<code>\BindingGutter</code>	5mm	Binding gutter to use if enabled (<code>\ifBindingGutter</code>)
<code>\BodyColumns</code>	1	Number of column to use for body text
<code>\ColumnGutterRuleSkip</code>	0pt	Space to insert above a gutter rule
<code>\columnshift</code>	0pt	Space to insert before text columns (e.g. for marginal verses)
<code>\ExtraRMargin</code>	0pt	Extra space to take from the right of a column
<code>\FigCreditPadding</code>	2pt	Space to insert above and below figure credits
<code>\FontSizeUnit</code>	1bp	Multiplier such that 12 is the desired font size
<code>\IntroColumns</code>	1	Number of columns for introductory text
<code>\ISBNfontdim</code>	10pt	Font size of ISBN barcodes
<code>\lastnotclubpenalty</code>	10000	Club penalty for the last note across pages
<code>\lastnoteinterlinepenalty</code>	10000	interline penalty for the last note of the page
<code>\lastnoteparpenalty</code>	100	manual par penalty for the last note
<code>\lastnotewidowpenalty</code>	10000	Widow penalty for the last note across pages
<code>\minblackline</code>	0.25pt	Minimum line thickness for a black line (see <code>OverThinRule</code>)
<code>\mincolourline</code>	0.5pt	as above for a colour line.
<code>\minwhiteline</code>	0.5pt	as above for a white line (presumably against a dark background)
<code>\minfontsize</code>	5pt	Trigger warnings if any font is smaller than this.
<code>\marginnotesgap</code>	0pt	How far are margin notes from the text (6pt)
<code>\marginnoteswidth</code>	0pt	How wide to make margin notes including gap (1in+6pt)
<code>\NoteCallerSpace</code>	.2em	Space following the note caller
<code>\NoteCallerWidth</code>	1.1ex	Width to normalise note callers to
<code>\NoteShaveMin</code>	1	Minimum number of lines of last note to split over
<code>\NoteShaveStay</code>	1	Minimum number of lines of all notes that remain
<code>\pageno</code>	1	Current page number
<code>\PaperHeight</code>	297mm	Page Height
<code>\PaperWidth</code>	210mm	Page width
<code>\StudyColumnGutterRuleSkip</code>	0pt	Space to insert above a study notes gutter rule
<code>\TabsEnd</code>	10pt	Distance between the lower edge of the lowermost thumb-tab and the bottom page margin (text area)
<code>\TabsStart</code>	10pt	Distance between the upper edge of the topmost thumb-tab and the top margin of the page (text area). Negative values may be given to extend tabs into the upper margin
<code>\TextBorderMode</code>	1	0 – border everything, 1 – exclude introductions, 2 diglots use full with introductions
<code>\TitleColumns</code>	1	Number of columns for a title block
<code>\VerseBoxWidth</code>	0pt	Allocated space for marginal verses (for wrapping)
<code>\XeTeXgenerateactualtex 0</code>		If non zero outputs actual text into the PDF file
<code>\XeTeXinterchartokensta 0</code>		If non zero, enables intercharacter space insertion
<code>te</code>		
<code>\XeTeXinterwordspacesh 0</code>		If non zero assume the primary font has contextual spaces (rare)
<code>aping</code>		
<code>\XeTeXuseglypmetrics 0</code>		If non zero use glyph metrics for box metrics

Dimension	Default	Description
\XrefNotesMargin	1pt	Margin between body text and column notes
\XrefNotesWidth	0pt	Width of central column notes
\XrefSkip	0pt	Insert space between cross references in centre column notes

Defined Constants

For \DefConstant, use \def\DefConstant{value}

Dimension	Default	Description
\AfterVerseSpaceFactor	2	Multiplied by \FontSizeUnit to insert after verse number in the text
\BackgroundOverrun	0.25pt	The spread beyond the text box when background colour set.
\BalanceThreshold	0.95	Don't let a full page become less full than this before taking the best unbalanced page
\BelowEndNoteRuleSpace	10pt	Space below and end note rule
\bold		Main bold body text font
\bolditalic		Main bold italic body text font
\BottomMarginFactor	\TopMarginFactor	\BottomMarginFactor * \MarginUnit space at the bottom of a page
\ChapPairSeparator	-	Range separator for chapters
\ChapterVerseSeparator	:	What to put between chapter and verse in reference
\colshiftmode	left	left, right, inner, outer for \columnshift
\DefaultSpaceBeside	10pt	Horizontal picture margin space * 2
\EndNoteRuleThickness	0.4pt	Thickness of end note rule
\EndNoteRuleWidth	0.5	Percentage of page width for end note rule
\FooterPosition	0.5	Multiplied by \MarginUnit. Position of baseline of footer from bottom of page
\FullPageFudgeFactor	0pt	How much does a full page top aligned picture miss the top of the page by?
\GraphPaperColMajor	0.8 0.1 0.8	Colour (RGB) of major division lines
\GraphPaperColMinor	0.9 1.0 1.0	Colour (RGB) of grid lines
\GraphPaperLineMajor	0.3pt	Thickness of major division lines
\GraphPaperLineMinor	0.1pt	Thickness of grid lines
\GraphPaperMajorDiv	5	How many divisions between major ones
\GraphPaperX	2mm	Grid size horizontally
\GraphPaperXoffset	0cm	Offset from start of page to start of grid
\GraphPaperY	2mm	Grid size vertically
\GraphPaperYoffset	0cm	Offset from start of page to start of grid
\HeaderPosition	0.5	Multiplied by \MarginUnit. Position of header baseline from top of page
\horizThumbtabVadj	1sp	Distance of top of tab box to text, unrotated. 1sp ⇒ centred
\IndentUnit	1in	Dimension to multiply indentation factors by
\intercharspace	\hskip0pt	Space to insert between characters if enabled.
\internotepenalty	-10	Penalty inserted between paragraphed notes
\InterNoteSpace	3.5pt	Vertical space between note classes
\ISBNfont	“Andika”	Font to use for text in ISBN
\italic		Main italic body text font
\LineSpaceBase	14	Effect font size of a default line spacing

Dimension	Default	Description
\LineSpacingFactor	1.0	FontSizeUnit multiplier for line spacing
\mainBodyColumns	1	How many columns for the main body text
\MarginNoteSide		One of "inner", "outer", "left", "right"
\margininversemarker		Auto generated verse range marker in margin verses. Styled with zmvm char style
\MaxPagesPerChunk	300	Spot runaway diglots by limiting pages per chunk
\OptionalBreakPenalty	300	Penalty for an optional break
\OverThinCurve	warn	What happens if a line is too thin for reliable printing
\OverThinRule	warn	(see \minblackline, \mincolourline, \minwhiteline) Options: "ignore" "warn" "warnfix" "fix" "error".
\PageFullFactor	0.9	Proportion of a page that is full before we say the page is full
\R[HF](noV title)?(odd even)(left center right)		Running header/footer tokens. noV for peripheral matter and introductions. title for pages that start with a major title.
\RangeChar	-	Character to use between references in a range
\RangeSeparator		What to put between first - last of a reference range
\regular		Main regular body text font
\RuleThickness	0.4pt	Thickness of a rule of not otherwise specified
\SideMarginFactor	1.0	\SideMarginFactor * \MarginUnit space either side of the page
\SmallCapsSuffix	/ICU:+smcp	What to add to font definition for \SmallCaps
\SpaceShrinkFactor	\empty	Dimension of inter character shrink space
\SpaceStretchFactor	\empty	Dimension of inter character stretch space
\StringOrnamentFont	Ag Arabesque Desktop	Font to use for ornament string definitions
\StudyGutterFactor	1.0	Gutter width factor for study notes
\StudyNoteRuleThickness	0.4pt	Rule above study notes thickness
\SuperscriptFactor	0.75	Font scale factor for \Superscript
\SuperscriptRaise	0.85ex	Default raise to use for \Superscript
\TabBleed	5pt	How much tabs protrude outside the page
\TextBorderColumns	{}	If not empty will put a text border around the specified diglot column(s) E.g. \def\TextBorderColumns{L}. See commit f2176832 log.
\TPILB	{}	Text to insert in intentional blank pages
\TopMarginFactor	1.0	\TopMarginFactor * \MarginUnit space at the top of a page
\UnderlineLower	0.2em	Distance of underline rule below lowest descender (if >=0), or below baseline (<0).
\UnderlineThickness	0.05em	Dimension of underline rule
\uselanguage{lang}	{}	Enable preinstalled language: USenglish, and a full list of language names from https://tug.org/texmf-dist/tex/generic/config/language.def . Lang tags: af as be bg bn ca cop cs cu cy da de-1901 de-1996 de-ch-1901 el-monoton el-polyton en-gb en-us eo es et eu fi fi-x-school fr fur ga gl gu hi hr hsb hu hy ia id it is ka kmr kn la-x-classic la-x-liturgic lt lv mk ml mn-cyrl mn-cyrl-x-lmc mr mul-ethi nb nl nn oc or pa pi pl pms pt rm ro ru sa sk sl sr-latn sr-cyrl sv ta te th tk tr uk zh-latn

Dimension	Default	Description
<code>\VerticalSpaceFactor</code>	1.0	Multiplier for SpaceBefore, SpaceAfter
<code>\vertThumtabVadj</code>	-2pt	Distance of top of tab box to text, rotated
<code>\vhyphenmode</code>		"bottom", "top", "left", "right" hyphen position in marginal verses
<code>\XrefNotes</code>		Notes class to use for central notes column
<code>\XrefSide</code>	0	1 = Left, 2 = Right, 3 = Inner, 4 = Outer

Other TeX file commands/options

TeX	Description
<code>\AllowAllTracing</code>	Allows tracing for user codes. Runs tracing slower
<code>\AutoCallers{x}{list}</code>	Comma separated (no spaces) list of callers for note x.
<code>\ColumnGutterFactor</code>	Column gutter width is <code>\ColumnGutterFactor * \FontSizeUnit</code> . Default 15.
<code>\marginaln@te{x}</code>	Specifies that x be a marginal note marker
<code>\NoStorePeriph{intot}</code>	Immediately output the peripheral section <i>intot</i> when it is met.
<code>\NoteAtEnd{x}</code>	Treat x as an endnote. <code>\fe</code> already set as endnote
<code>\NumericCallers{x}</code>	Specifies a note style that has numeric callers
<code>\OmitCallerInNote{x}</code>	Don't include a caller in this note
<code>\PageResetCallers{x}</code>	The callers for note style x should reset each page
<code>\ParagraphedNotes{x}</code>	Run notes in this class into one paragraph
<code>\StorePeriph{intot}</code>	Store the peripheral section <i>intot</i> for later inclusion using <code>\zgetperiph</code>
<code>\StudyNotes{x}</code>	Output x as study notes
<code>\NotStudyNotes{x}</code>	Don't output x as study notes
<code>\XeTeXlinebreaklocale</code>	ICU locale to use for line breaking
<code>\cstyle{sty}{text}</code>	Format text in character style sty.

PTX Penalty Values

Penalty	Default	Description
<code>OptionalBreakPenalty</code>	300	<code>\def</code> this to set <code>\linepenalty</code> at <code>//</code> . Also sets the break for a <code>//</code> as <code>-\OptionalBreakPenalty</code>
<code>badspacepenalty</code>	100	Assign this as the you can break here if you must

PTX Special Spaces and Characters

The default is `\def\intercharspace{\hskip 0pt}`

USV	Macro	Definition
00A0	<code>\NBSP</code>	<code>\nobreak\space</code>
00AD	<code>\SFTHYPHEN</code>	<code>\-</code>
2000	<code>\NQAD</code>	<code>\BADBREAK\relax\space</code>
2001	<code>\MQAD</code>	<code>\BADBREAK\hskip 1em plus .2em minus .2em</code>
2002	<code>\NSPACE</code>	<code>\hbox{\space}</code> (<i>non-stretching space</i>)
2003	<code>\MSPACE</code>	<code>\hskip 1em</code>
2004	<code>\THREEPEREMSPACE</code>	<code>\hskip .333em</code>
2005	<code>\FOURPEREMSPACE</code>	<code>\hskip .25em</code>
2006	<code>\SIXPEREMSPACE</code>	<code>\hskip .1666em</code>
2009	<code>\THINSPACE</code>	<code>\hskip .2em plus .1em minus .1em</code>
200A	<code>\HAIRSPACE</code>	<code>\hskip .042em\nobreak\intercharspace</code>
200B	<code>\ZWSP</code>	<code>\intercharspace</code>
2010	<code>\HYPHEN</code>	<code>\-</code>
2011	<code>\NBHYPHEN</code>	<code>\leavevmode\hbox{-}</code>
2028	<code>\LINESEP</code>	<code>\break</code>
2060	<code>\WJ</code>	<code>\leavevmode\nobreak</code>
2063	<code>\GOODBREAK</code>	<code>\penalty-\OptionalBreakPenalty</code>
2064	<code>\BADBREAK</code>	<code>\penalty\the\badspacepenalty</code>
FEFF	<code>\ZWNBS</code>	<code>\WJ</code>
	<code>\slash</code>	<code>/</code>

Hooks

Sethooks

These give the values for the first parameter (pos) of `\sethook{pos}{mrkr}{code}`

Position	Description
before	Runs before the style is set for the marker
start	Runs within the styling of the marker before any text is output
end	Runs before the marker closes or the paragraph is closed for a paragraph style. <i>For book-hooks (see below), it is at the end of the content, before any endnotes.</i>
after	Runs after the marker is closed and after the paragraph is finished for a paragraph style. <i>For book-hooks, this is after any end-of-book rule or pagebreak.</i>

‘mrkr’ can be any marker, or a complex marker such as ‘periph:intot|nd+is2’ Only hooks for the most complex matching hook position will be triggered. i.e. setting a start hook for `nd+s2` will prevent any hook defined for `nd` from triggering in an `s2` paragraph.

Other settable hooks:

<code>\setcvhook{#1}{code}</code>	Runs <i>code</i> at the start of the reference #1 e.g. GEN1.2 - At GEN 1:2 GEN3.0 - Just before chapter 3 of Genesis all3.0 - Just before chapter 3 in all books
<code>\setbookhook{pos}{GEN}{code}</code>	Runs <i>code</i> at start/end/after (of) GEN. If specified book is {all}, it applies to all books. {all} hooks are further from the text than {GEN} (etc) ones (i.e. start is earlier, end & final after).
<code>\sethook{bookstart}{GEN}{code}</code>	Alternative to <code>\setbookhook{start}</code>
<code>\setbookhook{pos}{final}{code}</code>	Runs <i>code</i> at end/final (of) the final book
<code>\sethook{final}{afterincludes}{code}</code>	Runs <i>code</i> after any included PDFs
<code>\sethook{page}{2}{code}</code>	Runs <i>code</i> just after page 1 has been written to the PDF.
<code>\setbetweenhook{mkr1}{mkr2}{code}</code>	Runs <i>code</i> between paragraphs mkr1 and mkr2. e.g. if it is desirable to insert extra space between <code>\s1</code> and <code>\r</code> but not above every <code>\r</code>

Note that if *code* is intended to set configuration variables with extended effect, it is probable that the prefix `\global` (or `\gdef`, `\xdef`) should be used. (‘probable’ becomes ‘certain’ for page hooks, start and end hooks on character styles and anything in a table, or heading)

AddHooks

These are additive, and should be used with extreme caution to avoid conflicts.

Addhook	Description
<code>\addtoinithooks</code>	Run when setting up on loading the first ptx file
<code>\addtoendhooks</code>	Run at the very end of the document after the last page
<code>\addtoeveryparhooks</code>	Run before the first character is added to a paragraph
<code>\addtoparstylehooks</code>	Run last at the end of a paragraph style
<code>\addtoidhooks</code>	Run when processing a <code>\id</code> line
<code>\addtoversehooks</code>	Run after the verse number is output

Addhook	Description
<code>\addtopreversehooks</code>	Run before the verse number is output
<code>\addToLeftHooks</code>	Run each <code>\lefttext</code> after entering the left column
<code>\addToRightHooks</code>	Run each <code>\righttext</code> after entering the right column
<code>\addToXHooks</code>	Run each <code>\polyglotcolumn X</code> , after entering that column
<code>\addToSideHooks{X}</code>	Alternative to <code>\addToXHooks</code>
<code>\addtoendptxhooks</code>	Run when closing a ptx file before end notes and single column
<code>\addtostartptxhooks</code>	Runs before a ptx file is read

Special hook macros

Various macros will be called if defined.

Macro	Description
<code>\hookchapter#1</code>	Called instead of outputting a dropped chapter number or label. Passed the chapter number.
<code>\note-no-insert-notetype</code>	Called instead of adding to an insert. Followed by content group
<code>\endnote-no-insert-notetype</code>	Called after the note content group closes
<code>\everyfirstverse#1#2#3</code>	#1=Book ID, #2=Book name from <code>\h</code> , #3=chapter number. Run before inserting verse number.
<code>\endheadingshook</code>	Called before outputting the <code>\he@dingbox</code>

Status tests / values for hooks etc.

(These may be tested to determine the state of things. As a debugging aid, the state of many of these, and some other internal variables are written to the log in response to meeting a `\SHOWSTUFF` in the USFM content).

Boolean/state	When valid	Why
<code>\iffilehasverses</code>	Any	False at the start of a file, set to true the first time that a <code>\v</code> has been met in a file.
<code>\ifm@rksonpage</code> <code>\ifstartpara</code>	Headings Text	Are there any verses on the about-to-be-printed page? True if a new paragraph has been initiated (e.g by <code>\p</code>) but it not actually begun yet (by some text)
<code>\prev@rsemode</code>	Text	Empty if the verse was mid-paragraph, equal to the paragraph type (e.g. 'q2') if the verse came at the start of the paragraph.
<code>\ifvmode</code>	Any	TeX has different modes. This tests for vertical mode when new boxes stack vertically, spaces are ignored and text starts a new paragraph and switches to horizontal mode. <code>\prevdepth</code> is valid
<code>\ifhmode</code>	Any	TeX has different modes. This tests for horizontal mode, when new boxes add horizontally. Spaces count. True in a paragraph or within an hbox.
<code>\prevdepth</code>	vertical mode	The depth below the baseline of the previous box on the page. Causes an error outside vertical mode.
<code>\iflastptxfile</code>	Any	Set true by template before the last file
<code>\ifinperiph</code>	Text	True when a <code>\periph</code> is being expanded
<code>\ifhe@dings</code>	Text	True when in a headings block

<code>\ifinextended</code>	Text	True when in an <code>\esb</code> (including automatically applied ones)
<code>\ifdoingt@ble</code>	Text	True when in a table.
<code>\ifinn@te</code>	Text	True when in a footnotes
<code>\iffirst@fterheading</code>	Text	Is the paragraph about to start the first one after a heading?
<code>\ifsk@pping</code>	Text	Is this due to be discarded? (e.g. because the current paragraph is a <code>\rem</code>)
<code>\iftextborder</code>	After <code>\id</code> (of last column, for <code>\diglot</code>)	True if text borders are configured for the main-body part of this book. Set false at the start of the book, set true (if appropriate) as part of <code>\id</code> processing.

When hooks aren't enough: complex layouts

In some cases, simply including a short piece of code / text before and after a run of text is insufficient or inelegant, additional fonts need selecting, or similar issues. For example `\rb` which is used in interlinears needs to select a font for the second lines. If a macro *complex-mrkr* exists, then that macro takes total control of the placement of the text and any other wrapping, etc, of it. The basic styled text is supplied in `\box0` (a horizontal box), and attributes supplied in the USFM are available via the `\get@ttribute{name}` method. 2 examples from the code are shown here:

```
\expandafter\def\csname complex-rb\endcsname{\rubyt{rb}{gloss}} % USFM-3
suggests ruby goes above. rubyb also exists
```

```
\def\rubyt#1#2{\trace{A}{rubyt}\setbox1=\hbox{\s@tfont{#2|#1}{#2|#1+
\styst@k}\op@ninghooks{start}{#2|#1}{#2|#1+}%
\get@ttribute{#2}\attr@b\cl@singhooks{end}{#2|#1}{#2|#1+}}%
\dimen0=\ifdim\wd0>\wd1 \wd0\else\wd1\fi
\setbox0=\hbox{\vbox{\s@tbaseline{#2|#1}{#2|#1+\styst@k}\hbox to
\dimen0{\alignb@x{#2|#1}{1}}%
\hbox to \dimen0{\alignb@x{#1}{0}}}}}
```

```
\expandafter\def\csname complex-ref\endcsname{%Process ref's loc
attribute by turning it into a GOTO link.
```

```
\ifRefLocLink
\get@ttribute{loc}%
\ifx\attr@b\relax\unhbox0\else
\pdfGOTO{\attr@b}\unhbox0\special{pdf:eann}%
\fi
\fi
}
```

Complex markers and the style stack

When the code needs to retrieve a style for a marker, such as `\s@tfont{MKR}{STK}`, `\s@tbaseline{MKR}{STK}` above, or more generally, `\getp@ram{parameter}{MKR}{STK}`, some searching is needed to determine what values should actually be in force. Normally two parameters (MKR and STK) are given to define the style. MKR is the marker (mostly ignored except for some definitions and error-checking), STK is the 'style stack', or marker-context, which is actually used, item by item, to find the relevant value to apply. If the code wants to look up an attribute, e.g. linespacing or fontcolour, then for example, when the stylestack is `rb+wj+p` (ruby text in words of Jesus, in a normal paragraph), it will check to see if `rb` defines the wanted parameter, then `wj`, then `p`. However, a stylesheet can define styling for a more specific location, based on all or part of the stack. Also there are prefixes like `id:GEN|`, any active `\cat` egories and milestones. Thus instead of looking up the style in `rb`, `wj`, and `p` it expands the lookup-list to actually look for all valid combinations of `id|GEN`, `rb`, `wj` and `p`, in order of how specific they are:

`id:GEN|rb+wj+p`, `rb+wj+p`, `id:GEN|rb+wj`, `rb+wj`, `id:GEN|rb`, `rb`, `id:GEN|wj+p`, `wj+p`,
`id:GEN|wj`, `wj`, `id:GEN|p`, `p`

(That expanded list is then cut down to what has a `\Marker` entry, and the shortened list cached until the end of the book).

Note that in the expansion (as illustrated above), the book (or other prefix) is considered less specific than the USFM context. This can be surprising if someone has defined `id:FRT|nd`, expecting it to apply to all of the front-matter, and then defines `nd+li1`, for a particular location elsewhere. If FRT includes use of `\nd` in a `li1` list context then the `nd+li1` styling will take precedence.

Also note that in defining complex styling elements, you may not skip levels. Styling defined for `nd+li1` will not be considered when the style stack is `nd+wj+li1`.

Dimension Units

Units	Description
pt	point (aka 'Printers' or 'American' point, defined in 1896)
pc	pica = 12pt
in	inch = 72.27pt
bp	big point (aka postscript point) 72bp = 1in
cm	centimeter: 2.54cm = 1in
mm	millimeter = 0.1cm
dd	didot point: 1157dd = 1238pt
cc	cicero = 12dd
sp	scaled point: 65536 sp = 1 pt. 1sp is smallest possible size TeX can calculate.

Layout Versions

Here we list the changes that were introduced at each layout version increase.

Test using `\ifversion{n}{new code}{old code}`. New code is for `\ptxversion == 0` or `>= n`.

Version	Description
2-ver 2.5	Subtract space below from space above+space below, thus decreasing the space above note rules. Fix asymmetric ISBN layout. Intro pages don't force the next book onto an odd page if the intro is missing. Don't increase linespacing for raised or lowered boxes. Don't shrink/stretch footnote text based on the paragraph the note is in. Default OrnamentUnit changes from font size to points.
3-ver 2.5.7	Change position of inline pictures. Don't indent firstline indent if less then the cutout width.
4-ver 2.6.6	Use range seperator for bridged verses. Trigger against in paragraph verses better. Only box char styles that really need it. Break columns down to including the depth of the text on the last line. Fix hanging verses with marginal verses. Split columns in page balancing more accurately. Count triggers better at the start of paragraphs.
5-ver 2.8.12	Better handling of IndentAtChapter and paragraph indenting or not. Better vertical positioning of marginal verse numbers. Fix manual page breaking.
6-ver 3.0	Use per note type shaving (between pages). Improve section head gridding. Add lower border padding.
7-ver 3.0.9	Don't carry over trigger references
8-ver 3.0.19	Move marks to after heading boxes
9-ver 3.0.31	Excludenotes from the available-height trial during diglot balancing

Tracing Codes

\tracing{x} where x is one of the codes shown below.

Code	Description	
a	Trace everything!	i inserts (tracking the value of holdinginserts)
A	Show attribute processing	l ifchecks
b	Give information on page balancing for external analysis	j paragraph adjustments (for seeing where it loses sync)
B	Underfull pages	m milestones
C	Show cutout calculations	mx milestones: expand content
c	Show chapters	M markers
ch	Active characters	n notes
cov	Covers	nS note saving / restoring
d	Diglot construction	o output routines / rebalancing
dP	Diglot page usage	oh other (not stylesheet) hooks
D	Diglot gory details	orn Ornaments
De	Diglot repetitive commands (each@col)	p page output
Df	Diglot footnote separation	pt Page tabs
Dh	Diglot available height extra details	P Piclists
Ds	Diglot stylesheet	s stylesheet
dm	Diglot module	sa stylesheet additional debugging
dmp	Diglot module: polyglot	spv stylesheet parameter lookups that return void
e	Extended USFM (mainly side bars and categories)	ss style start/stop
eb	Extended borders	sP Stylesheet everypar/chapter
ebi	Border style inheritance	sc Stylesheet category
et	textborders	sC Stylesheet Cache
f	footnotes	sh Stylesheet-related hooks
F	fonts	sk Style stack and option stack operations
g	figures	sko Style stack option-list building
G	shipping pages	S side-dependent swapping
gl	figure inserts - tracking saving / restoring of inserts	t Tables and table of contents
h	headings	T Triggers
hv	hanging verses	tim Timing various things.
H	Headers, marks, etc	u 'Unprintable' pages (e.g. triggered images don't fit)
		V verse numbers
		v marginal verses

Plugins

QR — generates QR codes (\zqrcode, \QRref)

polyglot-simplecols — single-page multi-column based polyglot layout

polyglot-complexpages — a more advanced multi-page based polyglot layout

TeX Info

Tracers

Property	Description
<code>\tracingcommands</code>	Show each command as it executes, =3 to see logic path
<code>\tracinglostchars</code>	Show all missing characters
<code>\tracingmacros</code>	Show each macro, its expansion and parameters
<code>\tracingonline</code>	Output <code>\showbox</code> to the terminal as well as the .log
<code>\tracingoutput</code>	Outputs the contents of each page to the log
<code>\tracingpages</code>	Show page break decision process
<code>\tracingparagraphs</code>	Show line breaking decision process
<code>\tracingrestores</code>	Trace removals from the 'save stack' for local variables
<code>\tracingstats</code>	Show resource usage stats at the end of the log
<code>\tracingifs</code>	Trace each <code>\if</code> , <code>\else</code> and <code>\fi</code>
<code>\tracingassigns</code>	When values are set.
<code>\tracinggroups</code>	Begin/end group scopes

Horizontal Penalties and Demerits

$\text{demerits} = \text{badness}^2 + \text{penalty}^2$ Set values: e.g. `\linepenalty=500`

Penalty	Default	Use
<code>linepenalty</code>	10/1000	Penalty for ending a line. Increase to reduce paragraph lengths.
<code>hyphenpenalty</code>	50	Penalty of inserting a discretionary break (hyphen) (10000 in headings)
<code>exhyphenpenalty</code>	50	Hyphen penalty if nothing is output
<code>XeTeXlinebreakpenalty</code>	0	Inserted at every XeTeXlinebreaklocale induced line break
<code>doublehyphendemerits</code>	5000	Extra cost to each line of multiple lines in sequence ending in a discretionary break
<code>finalhyphendemerits</code>	10000	Extra cost if the second-last line ends in a discretionary break
<code>adjdemerits</code>	10000	Extra cost if two lines have different tightnesses. I.e. don't let one line shrink and an adjacent one stretch.

Vertical Penalties

Excludes display and match penalties

Penalty	Default	Use
<code>clubpenalty</code>	150/10000	Extra penalty at the end of the first line of a paragraph
<code>widowpenalty</code>	150/10000	Extra penalty before the last line of a paragraph
<code>brokenpenalty</code>	100/50	Extra penalty for a line ending with a discretionary break.
<code>interlinepenalty</code>	0	Base penalty at the end of every line (10000 inside heading)
<code>outputpenalty</code>	10000	The penalty that caused the start of this page, or 10000
<code>insertpenalties</code>	/0	Penalty for adding an insert to a page
<code>floatingpenalty</code>	/500	Added to insertpenalties on insert split across pages.

Booknames

3 Ltr	2 Ltr	Name
GEN	GE	Genesis
EXO	EX	Exodus
LEV	LE	Leviticus
NUM	NU	Numbers
DEU	DT	Deuteronomy
JOS	JS	Joshua
JDG	JG	Judges
RUT	RU	Ruth
1SA	1S	1 Samuel
2SA	2S	2 Samuel
1KI	1K	1 Kings
2KI	2K	2 Kings
1CH	3K	1 Chronicles
2CH	4K	2 Chronicles
EZR	ER	Ezra
NEH	NE	Nehemiah
EST	ES	Esther
JOB	JB	Job
PSA	PS	Psalms
PRO	PR	Proverbs
ECC	EC	Ecclesiastes
SNG	SS	Song of Songs
ISA	IS	Isaiah
JER	JE	Jeremiah
LAM	LA	Lamentations
EZK	EZ	Ezekiel
DAN	DA	Daniel
HOS	HO	Hosea
JOL	JL	Joel
AMO	AM	Amos
OBA	OB	Obadiah
JON	JO	Jonah
MIC	MI	Micah
NAM	NA	Nahum
HAB	HB	Habakkuk
ZEP	ZP	Zephaniah
HAG	HG	Haggai
ZEC	ZE	Zechariah
MAL	MA	Malachi
MAT	MT	Matthew
MRK	MK	Mark
LUK	LK	Luke
JHN	JN	John
ACT	AC	Acts
ROM	RO	Romans
1CO	1C	1 Corinthians
2CO	2C	2 Corinthians
GAL	GA	Galatians

EPH	EP	Ephesians
PHP	PH	Philippians
COL	CL	Colossians
1TH	1T	1 Thessalonians
2TH	2T	2 Thessalonians
1TI	3T	1 Timothy
2TI	4T	2 Timothy
TIT	TT	Titus
PHM	PM	Philemon
HEB	HE	Hebrews
JAS	JA	James
1PE	1P	1 Peter
2PE	2P	2 Peter
1JN	1J	1 John
2JN	2J	2 John
3JN	3J	3 John
JUD	JD	Jude
REV	RE	Revelation
TOB	TO	Tobit
JDT	JT	Judith
ESG	EG	Esther Greek
WIS	WI	Wisdom of Solomon
SIR	SI	Sirach also Ecclesiasticus
BAR	BA	Baruch
LJE	LJ	Letter of Jeremiah
S3Y	S3	Prayer of Azariah and the Song of the Three Jews
SUS	SU	Susanna
BEL	BE	Bel and the Dragon
1MA	1M	1 Maccabees
2MA	2M	2 Maccabees
3MA	3M	3 Maccabees
4MA	4M	4 Maccabees
1ES	1E	1 Esdras (Greek)
2ES	2E	2 Esdras (Latin)
MAN	MN	Prayer of Manasseh
PS2	P2	Psalms 151
ODA	OD	Odes
PSS	P4	Psalms of Solomon
JSA	JH	Joshua A.
JDB	JI	Joshua B.
TBS	TB	Tobit S.
SST	ST	Susanna Th.
DNT	DN	Daniel Th.
BLT	BL	Bel Th.
3ES	3E	3 Ezra
EZA	EA	Apocalypse of Ezra
5EZ	5E	5 Ezra
6EZ	6E	6 Ezra

DAG	DG	Daniel (in Greek)	CNC	CC	Concordance
PS3	P3	Psalms 152-155	XXA	XA	Extra Material
2BA	2B	2 Baruch (Apocalypse)	XXB	XB	Extra Material
LBA	LB	Letter of Baruch	XXC	XC	Extra Material
JUB	JU	Jubilees	XXD	XD	Extra Material
ENO	EN	Enoch	XXE	XE	Extra Material
1MQ	1Q	1 Meqabyan/Mekabis	XXF	XF	Extra Material
2MQ	2Q	2 Meqabyan/Mekabis	XXG	XG	Extra Material
3MQ	3Q	3 Meqabyan/Mekabis	XXS	XS	Extra Material (not in PT)
REP	RE	Reproof (Proverbs 25-31)	BAK	BK	Back Matter
4BA	4B	4 Baruch (Rest of Baruch)	OTH	OT	Other Matter
LAO	LO	Letter to the Laodiceans	INT	IN	Introductory Peripherals
FRT	FR	Front Matter	TDX	TX	Topical Index
GLO	GL	Glossary / Wordlist	NDX	NX	Names Index

Regex

Special Characters

.	Matches any character except a newline.
^	Matches the start of the string.
\$	Matches the end of the string or just before the newline at the end of the string.
*	Matches 0 or more (greedy) repetitions of the preceding RE. Greedy means that it will match as many repetitions as possible.
+	Matches 1 or more (greedy) repetitions of the preceding RE.
?	Matches 0 or 1 (greedy) of the preceding RE.
*?,+?,??	Non-greedy versions of the previous three special characters.
*+,++,?+	Possessive versions of the previous three special characters.
{m,n}	Matches from m to n repetitions of the preceding RE.
{m,n}?	Non-greedy version of the above.
{m,n}+	Possessive version of the above.
{...}	Fuzzy matching constraints.
\\	Either escapes special characters or signals a special sequence.
[...]	Indicates a set of characters. A "^" as the first character indicates a complementing set.
	A B, creates an RE that will match either A or B.
(...)	Matches the RE inside the parentheses. The contents are captured and can be retrieved or matched later in the string.
(?flags-flags)	VERSION1: Sets/clears the flags for the remainder of the group or pattern; VERSION0: Sets the flags for the entire pattern.
(?:...)	Non-capturing version of regular parentheses.
(?>...)	Atomic non-capturing version of regular parentheses.
(?flags-flags:...)	Non-capturing version of regular parentheses with local flags.
(?P<name>...)	The substring matched by the group is accessible by name.
(?<name>...)	The substring matched by the group is accessible by name.
(?P=name)	Matches the text matched earlier by the group named name.
(?#...)	A comment; ignored.
(?=...)	Matches if ... matches next, but doesn't consume the string.
(?!...)	Matches if ... doesn't match next.
(?<=...)	Matches if preceded by
(?<!=...)	Matches if not preceded by
(?id)yes no	Matches yes pattern if group id matched, the (optional) no pattern otherwise.
(?(DEFINE)...)	If there's no group called "DEFINE", then ... will be ignored, but any group definitions will be available.
(?)	(? A B), creates an RE that will match either A or B, but reuses capture group numbers across the alternatives.
(*FAIL)	Forces matching to fail, which means immediate backtracking.
(*F)	Abbreviation for (*FAIL).
(*PRUNE)	Discards the current backtracking information. Its effect doesn't extend outside an atomic group or a lookahead.
(*SKIP)	Similar to (*PRUNE), except that it also sets where in the text the next attempt at matching the entire pattern will start. Its effect doesn't extend outside an atomic group or a lookahead.

Escapes

<code>\number</code>	Matches the contents of the group of the same number if number is no more than 2 digits, otherwise the character with the 3-digit octal code.
<code>\a</code>	Matches the bell character.
<code>\A</code>	Matches only at the start of the string.
<code>\b</code>	Matches the empty string, but only at the start or end of a word.
<code>\B</code>	Matches the empty string, but not at the start or end of a word.
<code>\d</code>	Matches any decimal digit ; equivalent to the set <code>[0-9]</code> when matching a bytestring or a Unicode string with the ASCII flag, or the whole range of Unicode digits when matching a Unicode string.
<code>\D</code>	Matches any non-digit character; equivalent to <code>[^\d]</code> .
<code>\f</code>	Matches the formfeed character.
<code>\g<name></code>	Matches the text matched by the group named name.
<code>\G</code>	Matches the empty string, but only at the position where the search started.
<code>\h</code>	Matches horizontal whitespace.
<code>\K</code>	Keeps only what follows for the entire match.
<code>\L<name></code>	Named list. The list is provided as a keyword argument.
<code>\m</code>	Matches the empty string, but only at the start of a word.
<code>\M</code>	Matches the empty string, but only at the end of a word.
<code>\n</code>	Matches the newline character.
<code>\N{name}</code>	Matches the named character.
<code>\p{name=value}</code>	Matches the character if its property has the specified value.
<code>\P{name=value}</code>	Matches the character if its property hasn't the value.
<code>\r</code>	Matches the carriage-return character.
<code>\s</code>	Matches any whitespace character ; equivalent to <code>[\t\n\r\f\v]</code> .
<code>\S</code>	Matches any non-whitespace character; equivalent to <code>[^\s]</code> .
<code>\t</code>	Matches the tab character.
<code>\uXXXX</code>	Matches the Unicode codepoint with 4-digit hex code XXXX.
<code>\UXXXXXXXX</code>	Matches the Unicode codepoint with 8-digit hex code XXXXXXXX.
<code>\v</code>	Matches the vertical tab character.
<code>\w</code>	Matches any alphanumeric character; equivalent to <code>[a-zA-Z0-9_]</code> when matching a bytestring or a Unicode string with the ASCII flag, or the whole range of Unicode alphanumeric characters (letters plus digits plus underscore) when matching a Unicode string. With LOCALE, it will match the set <code>[0-9_]</code> plus characters defined as letters for the current locale.
<code>\W</code>	Matches the complement of <code>\w</code> ; equivalent to <code>[^\w]</code> .
<code>\xxx</code>	Matches the character with 2-digit hex code XX.
<code>\X</code>	Matches a grapheme.
<code>\Z</code>	Matches only at the end of the string.
<code>\\</code>	Matches a literal backslash.

Flags

A	a	ASCII	Make <code>\w</code> , <code>\W</code> , <code>\b</code> , <code>\B</code> , <code>\d</code> , and <code>\D</code> match the corresponding ASCII character categories. Default when matching a bytestring.
B	b	BESTMATCH	Find the best fuzzy match (default is first).
D		DEBUG	Print the parsed pattern.

E	e	ENHANCEMATCH	Attempt to improve the fit after finding the first fuzzy match.
F	f	FULLCASE	Use full case-folding when performing case-insensitive matching in Unicode.
I	I	IGNORECASE	Perform case-insensitive matching.
L	L	LOCALE	Make \w, \W, \b, \B, \d, and \D dependent on the current locale. (One byte per character only.)
M	m	MULTILINE	"^" matches the beginning of lines (after a newline) as well as the string. "\$" matches the end of lines (before a newline) as well as the end of the string.
P	p	POSIX	Perform POSIX-standard matching (leftmost longest).
R	r	REVERSE	Searches backwards.
S	s	DOTALL	"." matches any character at all, including the newline.
U	u	UNICODE	Make \w, \W, \b, \B, \d, and \D dependent on the Unicode locale. Default when matching a Unicode string.
V0	V0	VERSION0	Turn on the old legacy behaviour.
V1	V1	VERSION1	Turn on the new enhanced behaviour. This flag includes the FULLCASE flag.
W	w	WORD	Make \b and \B work with default Unicode word breaks and make ".", "^" and "\$" work with Unicode line breaks.
X	x	VERBOSE	Ignore whitespace and comments for nicer looking REs.

Properties

Property Description

age	version character introduced
bc	Bidi Class , takes Bidi class values (AL, L, R, BN, etc.)
blk	Unicode block. E.g. Arabic_Ext_A, ASCII, etc.
ccc	Canonical combining class (or order)
dt	Decomposition type: Can, Com, Enc, etc.
ea	East Asian Width: A, F, H, N, Na, W
gc	General Category : L, Lo, Ll, M, Mc, Mn, P, etc.
GCB	Grapheme Cluster Break: SM, T, CN, etc.
hst	Hangul Syllable Type: L, LV, LVT, NA, T, V
InPC	Indic Positional Category : Bottom_And_Left, Left, etc.
InSC	Indic Syllabic Category : Bindu, Invisible_Stacker, Modifying_Letter, etc.
JSN	Jamo Short Name: A, AE, B, BB, BS, etc.
jjg	Joining Group: Beh, African_Feh, Heh_Goal, Malayalam_Ja, etc.
jt	Joining Type: C, D, L, R, T, U
lb	Line Break : AL, AL, B2, BA, BB, BK, CB, etc.
NFC_QC	NFC Quick Check: M(aybe), N(o), Y(es)
NFKC_QC	NFKC Quick Check: M, N, Y
nt	Numeric Type: De(cimal), Di(git), None, Nu(meric)
sc	Script : value is 4 letter script code in title case
SB	Sentence Break: AT, CL, CR, EX, FO, LE, LF, LO, NU, SC, SE, SP, ST, UP, XX
vo	Vertical Orientation: R, Tr, Tu, U
WB	Word Break: CR, DQ, Extend, FO, etc.
mkr ²	USFM marker tag (including \) categories: para, char, header, note, title, intro, table

²Not part of regular regex module. Only usable in changes.txt

Binary Properties

Property	Description
AHex	ASCII Hex Digit
Alpha	Alphabetic
Bidi_C	Bidi Control
Bidi_M	Bidi Mirrored
CI	Case Ignorable
Cased	Cased (has case)
CWCF	Changes When Casefolded
CWCM	Changes when Casemapped
CWL	Changes when lowercased
CWKCF	Changes when NFKCCasefolded
CWT	Changes when Titlecased
CWU	Changes when UPPERCASED
CE	Composition Exclusion
Dash	Is a Dash
DI	Default Ignorable Code Point
Dep	Deprecated
Dia	Diacritic
Emoji	Emoji
EComp	Emoji Component
EMod	Emoji Modifier
EBase	Emoji Modifier Base
EPres	Emoji Presentation
XO_NFC	Expands on NFC
XO_NFD	Expands on NFD
XO_NFKC	Expands on NFKC
XO_NFKD	Expands on NKKD
ExtPict	Extended Pictograph
Ext	Extender
Comp_Ext	Full Composition Exclusion
Gr_Base	Grapheme Base
Gr_Ext	Grapheme Extend
Gr_Link	Grapheme Link
Hex	Hex Digit
Hyphen	Hyphen
IDSB	Ideographic Descriptive Sequence Binary Operator

IDST	Ideographic Descriptive Sequence Trinary Operator
IDC	Ideographic Descriptive Sequence Continue
IDS	Ideographic Descriptive Sequence Start
Ideo	Ideographic
Join_C	Join Control
LOE	Logical Order Exception
Lower	Lowercase
Math	Math
NFD_QC	NFD Quick Check
NFKD_QC	NFKD Quick Check
NChar	Noncharacter Code Point
OAlpha	Other Alphabetic
ODI	Other Default Ignorable Code Point
OG_Ext	Other Grapheme Extend
OIDC	Other ID continue
OIDS	Other ID Start
OLower	Other Lowercase
OMath	Other Math
OUpper	Other Upper
Pat_Syn	Pattern Syntax
Pat_WS	Pattern Whitespace
PCM	Prepended Concatenation Mark
QMark	Quotation Mark
Radical	Radical
RI	Region Indicator
STerm	Sentence Terminal
SD	Soft Dotted
Term	Terminal Punctuation
UIDo	Unified Ideograph
Upper	Uppercase
VS	Variation Selector
WSpace	White Space
XIDC	XID Continue
XIDS	XID Start

Character Sets

See UTR#18. Binary properties take no value. All gc property values are treated as binary properties.

Character Set	Description
[charlist]	list of chars. Ranges separate first and last by '-'
[^charlist]	charset of non matching chars
\p{property}	charset containing all characters with the property
\p{property=value}	charset of all characters whose property is value
\P{property[=value]}	same as [^\p{property[=value]}]
[;property[=value]:]	legacy version of \P{property[=value]}

Character sets may be combined using relation operators:

Relation	Description
	union: $A \cup B$. Also absence of operator
&&	intersection: $A \cap B$
--	set difference: $A \setminus B$ (in A but not in B)
~~	symmetric difference: $A \oplus B = (A \cup B) \setminus (A \cap B)$

Printers

In the documentation above, there are various printer-compatibility measures:
\minblackline, \mincolourline, \minwhiteline, \minfontsize.
Line widths below the specified size can trigger a warning, a correction or an error; see
\OverThinRule (to specify the response for plain and double rules and borders) and
\OverThinCurve (for ornamental rules and borders). Oversmall fonts trigger a warning.

Pretore

Lines no thinner than 0.35pt. Diapositive (colour/grey) lines and cutouts at least 0.5pt
Black print qualities: low, medium, high
Bleeds to 3mm. Content should be 3mm from trim line.
Min page width: 85mm (3.34in)
There is no max page height (but keep usability in mind)

Maximum page width

Bandwidth	278mm (10.94in)	349mm (13.74in)	550mm (21.65in)
Max page width	133mm (5.24in)	160mm (6.29in)	255mm (10.04in)

Covers

Hard cover bleed: 20mm (0.78in)
Soft cover bleed: 3mm (0.118in)
Hard cover > 148x210mm. Diversion: 3mm (0.118in)
Hard cover < 148x210mm. Diversion: 2mm (0.078in)
Soft cover. Diversion: 0mm

Book Block Thickness

Max (comfortable) book block thickness is 5cm

Thinjet Paper Thickness

	ThinLeaf FSC		ThinOpaque white FSC			
Weight (gsm)	36	40	45	50	60	70
Thickness (µm)	44	49	55	61	73	86
Max no. of pgs (5cm)	2272	2040	1818	1638	1368	1162