

# OP-Bible – Technical Documentation

The code of the `op-bible.opm` macro file is described here.

`op-bible.opm`

```
3 \_codedec1 \processbooks {OpBible: macros for creating annotated Bible}
```

## 1 Preparatory work

Loading packages.

`op-bible.opm`

```
12 \_load[vlna] % single-letter prepositions and splitting hyphen managed specially in Czech
13 \_load[mte] % micro typographical extensions
14
15 \_namespace{opb}
```

Basic settings of  $\TeX$  parameters.

`op-bible.opm`

```
21 \_newdimen\lrmargin \lrmargin=10mm
22 \_margins/2 a4 (23,27,20,20)mm
23
24 \_typosize[11/13] % typesetting size of Bible text
25 \_hyperlinks\Blue\Blue % hyperlinks activated
26
27 \_parindent=20pt
28 \_nopagenumbers
29 \_mte_enablemte % micro typographical extensions enabled
30 \_vlna_singlechars {Czech}{AaIiVvOoUuSsZzKk} % lowercase "a" added to this family
31
32 \_showboxbreadth=0
33 \_let\notecolor=\Red
34
35 \_def\LightGrey {\_setcmymcolor{0 0 0 .1}}
36 \_def\LiRed {\_setcmymcolor{0 .2 .2 0}}
```

Fonts.

`op-bible.opm`

```
42 \_fontfam[lm]
43 \_fontfam[Heros] % fonts for notes
44 \_isfile{f-biblon.opm}\_iftrue
45 \_fontfam[biblon] % fonts for Bible text
46 \_else
47 \_let\Biblon=\LMfonts
48 \_fi
49
50 \_fontdef\bookfont{\_setfontsize{at19.pt}\_bf}
51 \_fontdef\chapfont{\_setfontsize{at13.pt}\_bf}
52 \_fontdef\markfont{\_setfontsize{at7pt}\_rm}
53 \_fontdef\captionfont{\Heros\cond\_setfontsize{at8pt}\_bf}
54 \_def\headfont{\Biblon\_setfontsize{at10pt}\_rm}
55 \_nsprivate \Biblon ;
```

Auxiliary macros. `\.printwarn`  $\{\langle text \rangle\}$  prints warning. `\.sedef`  $\{\langle name \rangle\}\{\langle body \rangle\}$  is expanded `\sdef`. `\.myaddto`  $\{\langle macro-name \rangle\}\{\langle text \rangle\}$  adds  $\langle text \rangle$  to  $\langle macro-name \rangle$  globally. Moreover it defines the undefined macro by `\sdef{\langle macro-name \rangle}\{\langle text \rangle\}`.

`op-bible.opm`

```
65 \_let\printwarn=\opwarning
66 \_def \.sedef #1{\_ea\_edef \_csname#1\_endcsname}
67 \_long\_def\myaddto#1#2{\_ifcsname#1\_endcsname
68 \_gobal\_ea\_addto\_csname#1\_endcsname#2\_else \_global\sdef{#1}{#2}\_fi}
```

## 2 The main loop over Bible books

The `\processbooks` macro does two loops over all marks in `\printedbooks`. The macro `\printedbooks` is a list of `<a-marks>` of Bible books separated by spaces and it must be defined in the main file. The `\_useit` trick is used here in order we want to add `<space>{}` at the end of the expanded `\printedbooks`. The first loop body sets `\pbook!<a-mark>` used for hyperlinks. The second loop body does:

- Defines `\amark` as `<a-mark>` (an actual mark of the book used in the text).
- Defines `\bmark` as `<b-mark>` (a mark of the book used in file names).
- Defines `\.btit` as the book title.
- Saves `<a-mark>` to the `\.currbook` macro.
- Calls `\.newbook{<a-mark>}`
- Prints title of the book to the terminal and to the log.
- Calls `\bex!<a-mark>` in order to apply the `\BookException` data.
- Inputs introduction file if it exists. The real `\input` and formatin of the introduction text is done by the `\.printintro` macro.
- Inputs format definition file if it exists. Information is saved to the T<sub>E</sub>X memory.
- Inputs notes file if it exists. The notes are saved to the T<sub>E</sub>X memory.
- Calls `\bpr!<a-mark>` in order to apply the `\BookPre` data.
- Inputs txs file with original text of the Bible using `\.bibleinput`, i.e. prints the text from txs file with notes from the T<sub>E</sub>X memory.
- Calls `\bpo!<a-mark>` in order to apply `\BookPost` data.

Note that the macros `\introfile`, `\fmtfile`, and `\notesfile` give the location of appropriate files and these macros must be defined by the user in the main file.

Note2: each book of the Bible is processed in the group. It means that all data from notes, formats etc. are stored in the memory only temporary for processing single book. After the Book is finalized, the T<sub>E</sub>X memory is freed.

op-bible.opm

```

108 \_def\processbooks {\_par
109   \checknochapbooks
110   \_useit{\_ea\processbooksA \printedbooks} {}
111   \_useit{\_ea\processbooksB \printedbooks} {}
112 }
113 \_def\processbooksA #1 {%
114   \_if\_relax#1\_relax \_else \_sxdef{pbook!#1}{\_ea\processbooksA \_fi
115 }
116 \_def\processbooksB #1 {%
117   \_if\_relax#1\_relax \_else
118     \_edef\amark{#1}
119     \_edef\bmark{\_cs{f!#1}}
120     \_edef\.btit{\_cs{btit!#1}}
121     \_begingroup
122       \_edef\.currbook{#1}
123       \.newbook{#1}
124       \_wterm{** \_cs{btit!#1} {#1} **}
125       \_cs{bex!#1}
126       \_isfile{\introfile}\_iftrue \.printintro
127       \_else \.printwarn{File with introduction text \introfile\_space not found}\_fi
128 %
129       \_isfile{\fmtfile}\_iftrue \_input{\fmtfile}
130       \_else \.printwarn{File with format info \fmtfile\_space not found}\_fi
131       \_isfile{\notesfile}\_iftrue \_input{\notesfile}
132       \_else \.printwarn{File with notes \notesfile\_space not found}\_fi
133       \_cs{bpr!#1}
134       \.bibleinput{txsfile}
135       \_cs{bpo!#1}
136     \_endgroup
137     \_ea \processbooksB
138   \_fi
139 }
140 \_nspublic \processbooks ;

```

`\.newbook{<a-mark>}` ejects previous page, prepeares header and prints the book title.

```

146 \_def\newbook#1{\_vfil\_supereject
147 \_let\prelinkB=\currbook \_chapnum=0
148 \_def\prelinkC{0}\_def\prelinkV{0}
149 \_global\_headline={\_hfil \_ea\setheadline\_ea{\_btit}}
150 \_linef{\_hss\bookfont\btit\_hss}
151 \_par\nobreak\_medskip
152 }

```

`\setheadline{<book-title>}` sets `\_headline`. It is re-set for each new book by `\newbook`.

The `\bibname` can be defined by user as a name of the translating variant of the Bible. If it is not defined then it is empty by default.

```

161 \_def\setheadline#1{\_global\_headline={\_headfont
162 \_ifodd\_pageno
163 \_rlap{\_it\bibname\_hss}%
164 \_hfil \_the\_pageno\_hfil
165 \_hbox to\lrmargin{\_hss\_bf#1\_ifx^\_botmark^\_else\_space \_botmark\_fi}%
166 \_kern-\lrmargin
167 \_else
168 \_kern-\lrmargin
169 \_hbox to\lrmargin{\_bf#1 \_firstmark\_hss}%
170 \_hfil \_the\_pageno\_hfil
171 \_llap{\_hss\_it\bibname}%
172 \_fi
173 }
174 }
175 \_def\bibname{}

```

We want `<Fm 4>` to be a link to `Fm/1:4` because it is a single-chapter book. Compare `<Gn 4>` which is a link to `Gn/4:1`. There is a list of single-chapter books `\nochapbooks`. User must define it. The marks of these single-chapter books are separated by spaces here. The first and the last space are added to the `\nochapbooks` macro because we need them in `\.brefBookChapter`. The `\.checknochapbooks` macro does it, moreover, it checks if the `\nochapbooks` is defined. If not, it prints warning.

```

188 \_def\checknochapbooks {%
189 \_ifx\nochapbooks\_undefined
190 \.printwarn{\_noexpand\nochapbooks (boks without chapters) undefined.}%
191 \_def\nochapbooks{}%
192 \_else \_edef\nochapbooks{\_space\nochapbooks\_space}\_fi
193 }

```

### 3 Book titles

The macro `\BookTitle <a-mark> <b-mark> {<title>}` declares titles of each Bible books. The `<a-mark>` is an actual book mark used in printed text. The `<b-mark>` can be used in file names as `\bmark`. The mapping is done here: `\def\btit!<a-mark>{<title>}`, `\def\bf!<a-mark>{<b-mark>}`.

The macro is defined as `\outer` because we don't want to see obscure errors due to missing a space after `<b-mark>` or `<a-mark>`.

```

210 \_outer\_def\BookTitle #1 #2 #3{\_sxdef{btit!#1}{#3}\_sxdef{f!#1}{#2}}

```

The `\BookException <a-mark> {<code>}` macro adds the `<code>` to the `\bex!<a-mark>` macro. It is used in `\processbooks` loop in the group before files are read. You can redefine some filenames or something more special here.

Macros `\BookPre <a-mark> {<code>}` and `\BookPost <a-mark> {<code>}` are defined similarly. They add `<code>` to the `\bpr!<a-mark>` and to the `\bpo!<a-mark>` macros respectively.

```

222 \_outer\_long\_def\BookException #1 #2{\.myaddto{bex!#1}{#2}}
223 \_outer\_long\_def\BookPre #1 #2{\.myaddto{bpr!#1}{#2}}
224 \_outer\_long\_def\BookPost #1 #2{\.myaddto{bpo!#1}{#2}}
225
226 \_nspublic \BookTitle \BookException \BookPre \BookPost ;

```

## 4 Actions

We create the output in two steps. First step: the data from `\Note` etc. are read and saved to the T<sub>E</sub>X memory. For each such data element the “action” is registered to a list of actions of the given verse. Each Bible verse has its list of actions. The second step: the Bible verses are read from a `.txs` file and all appropriate actions (registered to this verse) are processed before the verse text is printed. These actions can modify the selected parts of the verse text.

`\alist!` $\langle full-vref \rangle$  is the list of actions associated with the verse  $\langle full-vref \rangle$ . The  $\langle full-vref \rangle$  is full reference to the verse in the format  $\langle book-mark \rangle / \langle chapter-num \rangle : \langle verse-num \rangle$

`\.newaction` $\{\langle full-vref \rangle\} \{ \langle action-body \rangle \}$  allocates new action.

op-bible.opm

```
246 \_def\.newaction#1#2{%
247   \_unless\_ifc\name alist!#1\_endcsname \_sdef{alist!#1}{\_fi
248   \_ea\_addto\_csname alist!#1\_endcsname{#2}%
249 }
```

A typical “action” is `\.replpre`. The actions are processed for each Bible verse when the verse text is saved to the `\.buff` macro. The `\.buff` macro is processed after all actions of given verse are done.

`\.replpre` $\{\langle prefix \rangle\} \{ \langle text \rangle \} \{ \langle fail \rangle \}$  replaces first occurrence of  $\langle text \rangle$  by  $\langle prefix \rangle \{ \langle text \rangle \}$  in `\.buff` macro. If the  $\langle text \rangle$  is empty then  $\langle prefix \rangle \{ \}$  is inserted at the beginning of the `\.buff`.

If  $\langle text \rangle$  does not exists then  $\langle fail \rangle$  is processed. It can report failed  $\langle text \rangle$  by the `\.text` macro.

op-bible.opm

```
262 \_def\.replpre#1#2#3{%
263   \_ifx~#2\_def\.tmp{#1}{\_ea\_ea\_ea\_def\_ea\_ea\_ea\.buff\_ea\_ea\_ea{\_ea\.tmp\.buff}%
264   \_else
265     \_def\.replpredo##1#2##2\_end{%
266       \_ifx~##2\_def\.text{#2}#3% <fail>
267       \_else \.replsave ##1#1{#2}##2\_end \_fi
268     }%
269     \_def\.replsave##1#2\_end{\_def\.buff{##1}}%
270     \_ea\.replpredo\.buff#2\_end
271   \_fi
272 }
```

## 5 The \Note macro

The first parameter of the `\Note` macro is  $\langle gen-vref \rangle$ . It is generalized reference to the Bible verse. It can be  $\langle chapter-num \rangle : \langle verse \rangle$  (the  $\langle book-mark \rangle$  is appended from the `\.currbook` macro) or  $\langle chapter-num \rangle : \langle verse-from \rangle - \langle verse-to \rangle$  (only  $\langle verse-from \rangle$  is used for generating  $\langle gen-vref \rangle$ ).

`\.gentovref` $\{\langle gen-vref \rangle\}$  expands to  $\langle full-vref \rangle$ .

op-bible.opm

```
286 \_def\.gentovref#1{\.currbook/\.gentovrefA#1-\end}
287 \_def\.gentovrefA#1-#2\end{#1}
```

`\.renumvref`  $\langle full-vref \rangle \backslash\_relax$  does re-calculating of  $\langle full-vref \rangle$  using `\renum` data.

op-bible.opm

```
294 \_def\.renumvref #1/#2\_relax{#1/\_trycs{rn!\tmark!#1/#2}{#2}}
```

The  $\langle word \rangle$  given as a parameter of the `\Note` macro (see bellow) is used as a word phrase which should be searched in the given verse text. This parameter  $\langle word \rangle$  is transformed first by expansion of `\.transformword` $\{\langle word \rangle\}$  to the  $\langle tword \rangle$  variant and the  $\langle tword \rangle$  is actually used for searching. The `\.transformword` $\{\langle word \rangle\}$  expands to the variant of the  $\langle word \rangle$  declared by `\.vdef`. If not declared then it expands to the  $\langle word \rangle$  itself, i.e  $\langle tword \rangle$  is equal to  $\langle word \rangle$  in this case.

op-bible.opm

```
305 \_def\.transformword#1{%
306   \_ifc\name v!\tmark!#1\_endcsname \_lastnamedcs
307   \_else #1\_fi
308 }
```

`\Note`  $\langle gen-vref \rangle \langle space \rangle \{ \langle word \rangle \} \langle text \rangle \backslash\_par$  transforms  $\langle word \rangle$  to the  $\langle tword \rangle$  (see above), saves  $\langle text \rangle$  and activates replace-action of  $\langle tword \rangle$  to `\.doNote` $\{\langle note-num \rangle\} \{ \langle tword \rangle \}$  in given verse.

There is an alternative syntax `\Note` $\langle gen-vref \rangle \langle space \rangle \{ \langle word \rangle \} = \{ \langle pword \rangle \} \langle text \rangle \backslash\_par$  If  $\langle pword \rangle$  is given then it is printed in the note instead  $\langle tword \rangle$ . More precisely: transformed  $\langle word \rangle$  is used for searching (and it is kept in the verse unchanged) but  $\langle pword \rangle$  is printed in the note.

The `\ww` can precede `\Note`. If it is true then the  $\langle word \rangle$  is prepared in `\nextww` and  $\langle pword \rangle$  is in `\nextwwA`. Otherwise, the macros `\nextww` and `\nextwwA` are undefined.

`\Note` does exactly following:

- Calculates  $\langle full-vref \rangle$  using  $\backslash.gentovref\{\langle gen-vref \rangle\}$  and svese it to  $\backslash.fullvref$ .
- If the verse number of  $\langle full-vref \rangle$  is zero, we want to insert the note-text before the chapter. This is one by the  $\backslash.NoteB$  macro.
- Allocates new  $\langle note-num \rangle$ , i.e.  $\backslash.notenum$  is  $\langle note-num \rangle$ .
- Modifies  $\langle full-vref \rangle$  if  $\backslash.renum$  was declared using  $\backslash.renumvref$  and saves the result to  $\backslash.fullvrefm$ .
- Uses  $\backslash.nexttw$  and  $\backslash.nexttwA$  as  $\langle tword \rangle$  and  $\langle pword \rangle$  if they are defined.
- Otherwise transforms  $\langle word \rangle$  to  $\langle tword \rangle$  by  $\backslash.transformword$ .
- Reads  $\langle pword \rangle$  (word to be printed in the note) by  $\backslash.NoteA$  if the alternative syntax with  $=\{\langle pword \rangle\}$  is used. Else  $\langle pword \rangle$  is equal to  $\langle tword \rangle$ . Use it only if  $\backslash.nexttw$  is undefined.
- Defines  $\backslash.notetext!\langle note-num \rangle$  as  $\langle text \rangle$ .
- Defines  $\backslash.noteref!\langle note-num \rangle$  as  $\langle full-vref \rangle$  re-calculated by  $\backslash.renum$ .
- Defines  $\backslash.notepre!\langle note-num \rangle$  as numeric part of modified  $\langle full-vref \rangle$ . and calculates  $\langle from \rangle$ - $\langle to \rangle$  part (if exists in  $\langle gen-vref \rangle$ ) using  $\backslash.renumlabel$  macro. This is printed prefix of the  $\backslash>Note$ .
- Defines  $\backslash.pword!\langle note-num \rangle$  as  $\langle pword \rangle$ ,
- Does  $\backslash.newaction\{\langle full-vref \rangle\}\backslash.replpre\{\backslash.doNote\{\langle note-num \rangle\}\}\{\langle tword \rangle\}\backslash.notefail\{\langle note-num \rangle\}\}$ .

This is done by  $\backslash.AddNote\{\langle full-vref \rangle\}\{\langle note-num \rangle\}\{\langle tword \rangle\}$ .

Note that  $\backslash>Note$  is defined as  $\backslashouter$  in order to report correctly typical mistakes with missing empty line the text of a previous  $\backslash>Note$ .

op-bible.opm

```

354 \_newcount\_.notenum
355 \_def\_.Note #1 #2{%
356   \_edef\_.fullvref{\_.gentovref{#1}}%
357   \_ea\_.isversezero\_.fullvref\_iftrue
358     \_ea\_.NoteB
359   \_else
360     \_incr\_.notenum
361     \_edef\_.fullvrefm{\_ea\_.renumvref\_.fullvref\_relax}%
362     \_def\_.tmp{#1}\_.sedef{notepre!\_the\_.notenum}\{\_ea\_.renumlabel\_.fullvrefm\_relax}%
363     \_ifx\_.nexttw\_undefined
364       {\_def\_.printwarn##1{\_xdef\_.tword{\_.transformword{#2}}}%
365       \_else \_xdef\_.tword{\_.nexttw}\_fi
366       \_afterfi{\_isnextchar={\_.NoteA}\{\_.NoteA={}\}}%
367     \_fi
368 }
369 \_def\_.NoteA=#1#2% #2 separated by \par or \_par:
370
371 {%
372   \_sdef{notetext!\_the\_.notenum}\{\_ignorespaces#2}%
373   \_.sedef{noteref!\_the\_.notenum}\{\_.fullvrefm}%
374   \_ifx\_.nexttw\_undefined
375     \_ifx^#1^\_sdef{pword!\_the\_.notenum\_ea\_ea\_.tword}\_else \_sdef{pword!\_the\_.notenum}\{#1}\_fi
376   \_else
377     \_sdef{pword!\_the\_.notenum\_ea\_ea\_.nexttwA}%
378     \_let\_.nexttw=\_undefined \_let\_.nexttwA=\_undefined
379   \_fi
380   \_ea\_.addNote\_expanded{\_.fullvrefm}\{\_the\_.notenum}\{\_.tword}\}%
381 }
382 \_def\_.addNote#1#2#3{%
383   \_ifx^#3^ \_.tword is empty
384     \_.newaction{#1}\{\_addto\_.prebuff{\_.doNote{#2}\}\}%
385   \_else
386     \_.newaction{#1}\{\_.replpre{\_.doNote{#2}\}\{#3\}\_.notefail{#2}\}%
387   \_fi
388 }
389 %\_outer\_def\_.Note{\_.Note} % will be done at the end of this macro file

```

The  $\backslash.NoteB \langle text \rangle \backslash par$  does not register any action to the verse but defines  $\backslash chapnote!\langle full-vref \rangle$  as the  $\langle text \rangle$ . This chapter note will be printed before the chapter starts.

op-bible.opm

```

398 \_def\_.NoteB #1% #1 separated by \par or \_par
399
400 {%
401   \_sdef{chapnote!\_.fullvref}\{\_ignorespaces#1}%
402 }
403 \_def\_.isversezero#1/#2:#3\_iftrue{\_ifnum #3=0 }

```

`\.renumlabel`  $\langle full-vref \rangle$  `\_relax` expands to the numeric part of  $\langle full-vref \rangle$  and appends the `-- $\langle to \rangle$`  part if the `\.tmp` macro is in the format  $\langle chapter \rangle:\langle from \rangle-\langle to \rangle$ . The  $\langle to \rangle$  part is re-calculated in order to the number of verses between  $\langle from \rangle$  and  $\langle to \rangle$  be kept. If the  $\langle to \rangle$  part is in the format  $\langle chapter \rangle:\langle verse \rangle$  then it is unchanged. The `\.renumlabel` macro must be expandable, so we cannot use `\isinlist` and we prepare special expandable macros `\.isdivisin` and `\.iscolonin`.

op-bible.opm

```
416 \_def\.renumlabel#1/#2\_relax#2%
417 \_ea\.isdivisin\.tmp-\_iftrue --\_ea\.renumlabelA\.tmp\_relax#2\_relax \_fi
418 }
419 \_def\.renumlabelA#1:#2-#3\_relax#4:#5\_relax{%
420 \.iscolonin#3:\_iftrue #3\_else \_the\_numexpr#5+#3-#2\_relax \_fi
421 }
```

The `\Note` text is processed and printed in the second step, when the `.txs` file is read. Actions are assigned to each verse and they are run before the appropriate verse is printed. And `\Note` action says:

```
\.replpre{\.doNote{<note-num>}}{<tword>}{\.notefail{<note-num>}}
```

It means that the  $\langle tword \rangle$  is searched in the verse text and replaced by `\.doNote{ $\langle note-num \rangle$ }{ $\langle tword \rangle$ }`. If  $\langle tword \rangle$  is not found then `\.notefail{ $\langle note-num \rangle$ }` prints warning about it and `\.doNote{ $\langle note-num \rangle$ }{}` is prefixed before the verse text.

op-bible.opm

```
436 \_def\.notefail#1{%
437 \.printwarn{\_csstring\\Note: \.currverse: The text "\_unexpanded\_ea{\.text}" not found}%
438 \.replpre{\.doNote{#1}}{}}{\_Note is registered with the beginning of the verse
439 }
```

The `\.doNote{ $\langle note-num \rangle$ }{ $\langle tword \rangle$ }` prints the real note text in the second step, when the verse text from `\.buff` is processed.

op-bible.opm

```
446 \_def\.prevtmpb{}
447 \_def\.doNote#1#2{%
448 \_edef\.tmpb{\_cs{notepre!#1}}%
449 \.notelog{\_space\_space \_csstring\\Note \.tmpb\_space {#2}={\_cs{pword!#1}} {#1}}%
450 \.noteinsert{%
451 {\_bf \_ifx\.prevtmpb\.tmpb \_else \.tmpb \_enskip \_glet\.prevtmpb=\.tmpb \_fi
452 \.trymakedest{n:\_cs{noteref!#1}}%
453 \_ea\_ifx \_csname pword!#1\_endcsname \_empty
454 \_else \_ea\_ea\_ea\.upcasefirst \_csname pword!#1\_endcsname. \_fi}%
455 \_cs{notetext!#1}}%
456 {\_notecolor#2}%
457 }
458 \_def\_printfnotemark{}
459 \_def\_textindent#1{\_noindent}
```

The phrase  $\{\langle word \rangle\}$  used in notes must be exactly the same as the word used in the `.txs` text. But we want to capitalize the first letter of the  $\langle word \rangle$  when printing by `\.upcasefirst`. You can say `\let\.upcasefirst=\relax` if you don't want this feature.

op-bible.opm

```
469 \_def\.upcasefirst #1{\_uppercase{#1}}
```

Because there is asynchronous processing of the `\Note` text, we have a problem when an error occurs here. We cannot reference to appropriate line where the `\Note` is written. So, we print the parameters of processed `\Note` to the log file. The user can look into this file and the last printed `\Note` parameters here refers probably to the `\Note` where the reason of the error is.

The logging is done by `\.notelog{ $\langle text \rangle$ }`. It is `\wlog` by default but you can set it to `\ignoreit` or `\wterm`.

op-bible.opm

```
482 \_let\.notelog=\_wlog
```

## 6 Inserting data from format files

`\fmtpre`  $\{\langle gen-vref \rangle\}\{\langle what \rangle\}$  adds  $\langle what \rangle$  to `\.fmtprebuff`, i.e. at the beginning of the verse.

`\ftmadd`  $\{\langle gen-vref \rangle\}\{\langle what \rangle\}$  adds  $\langle what \rangle$  to `\.buff`, i.e. at the end of the verse.

`\fmtins`  $\{\langle gen-vref \rangle\}\{\langle text \rangle\}\{\langle what \rangle\}$  inserts  $\langle what \rangle$  after  $\langle text \rangle$  in the verse. If  $\langle text \rangle$  is not found then

$\langle what \rangle$  is inserted like `\fmtpre` does it  
All these commands allocate new action using `\.newaction`.

op-bible.opm

```
497 \_def\.fmtpre#1#2{\.newaction{\.gentovref{#1}}{\_addto\.fmtprebuff{#2}}}
498 \_def\.fmtadd#1#2{\.newaction{\.gentovref{#1}}{\_addto\.buff{#2}}}
499 \_def\.fmtins#1#2#3{\.newaction{\.gentovref{#1}}{\.replpre{\.fmtafter{#3}}{#2}{\.fmtfail{#3}}}}
500 \_def\.fmtafter#1#2{#2#1}
501 \_def\.fmtfail#1{\.fmtwarn\_addto\.fmtprebuff{#1}}
502 \_def\.fmtwarn{\.printwarn{\_stringfmtins: \.currverse: The text "\.text" not found}}
503
504 \_nspublic \fmtpre \fmtadd \fmtins ;
```

`\begcenter` starts the centering mode. It opens a group and does setting. User must use paired `\endcenter` in order to close this group. The `\centeringmode` status is checked by `\encenter` because curious error (about # character) should be occur without this checking.

op-bible.opm

```
513 \_newdimen\centermargin \centermargin=4em
514 \_def\.begcenter{\_par \_ifnum\_lastpenalty<10000 \_medskip \_fi
515   \_bgroup
516   \_def\.centeringmode{y}
517   \_parindent=0pt
518   \_leftskip=\centermargin plusfilll
519   \_rightskip=\leftskip
520 }
521 \_def\.endcenter{\_par
522   \_ifx\.centeringmode\_undefined
523   \_printwarn{\_noexpand\endcenter ignored: no \_noexpand\begcenter precedes}
524   \_else \_egroup \_medskip \_fi
525 }
526 \_nspublic \begcenter \endcenter ;
```

## 7 Printing verses from .txs files

When Bible text is processed then book mark is saved to `\.currbook` and each input line is separated to the  $\langle chapter-num \rangle$ : $\langle verse-num \rangle$  and  $\langle verse-text \rangle$ .

The `\.processline`  $\langle chapter \rangle$ : $\langle verse \rangle$  $\langle space \rangle$  $\langle verse-text \rangle$ ~J is repeatedly processed.

op-bible.opm

```
539 \_eoldef\.processline#1{\.processverse \.currbook/#1\_end}
```

`\.processverse`  $\langle full-vref \rangle$  $\langle space \rangle$  $\langle verse-text \rangle$ ~J does

- defines `\.currverse` as  $\langle full-vref \rangle$ ,
- prepares `\.currversenum`, `\.currversetext`, `\.currchapnum` from  $\langle full-vref \rangle$ ,
- defines `\.buff` as  $\langle verse-text \rangle$ ,
- processes all actions from `\alist!` $\langle full-vref \rangle$ ,
- if `\.currchapnum` changed, prints new chapter by `\.printchap`
- prints verse from `\.buff` using `\.printverse`

op-bible.opm

```
553 \_newcount\chapnum
554 \_def\.processverse #1 #2\_end{%
555   \_edef\.currverse{#1}%
556   \_preparechapverse #1
557   \_let\prelinkV=\.currversenum
558   \_def\.buff{#2}\_def\.fmtprebuff{\_def\prebuff{}}%
559   \_ifx\verseto\_empty \_csname alist!#1\_endcsname \_else
560     \_fornum\versefrom..\verseto \_do{\_csname alist!\.currbook/\.currchapnum:##1\_endcsname}%
561   \_fi
562   \_ifnum\.currchapnum=\chapnum \_else
563     \_let\prelinkC=\.currchapnum \chapnum=\.currchapnum\_relax \.printchap \_fi
564   \.printverse
565 }
566 \_def\preparechapverse #1/#2:#3 {\_def\.currchapnum{#2}%
567   \_def\verseto{}}%
568   \.isdivisin #3-\_iftrue \.defversefromto #3\_end
569   \_else \_def\.currversenum{#3}\_let\.currversetext=\.currversenum
570   \_fi
```



```

571 }
572 \_def\defversefromto #1-#2\_end{%
573   \_def\versefrom{#1}\_def\verseto{#2}%
574   \_def\currversenum{#1}\_def\currversetext{#1--#2}}

```

`\.printverse` prints verse from `\.currversenum` and (possibly changed) `\.buff`. It prints the single raised verse number first.

`\.printchap` prints beginning of the new chapter. `\.printbeforefirst` is a macro which is executed just before first verse of the chapter, after all material from `\fmtpre` is executed. I.e after printing a chapter name (if declared by `\fmtpre`).

op-bible.opm

```

585 \_def\.printverse{%
586   \fmtprebuff % material accumulated by \fmtpre
587   \_ifnum\.currversenum=1 \.printbeforefirst \_fi
588   \_quitvmode \_mark{\.currchapnum:\.currversetext}%
589   \_ifx\verseto\_empty \_trymakedest{v:\.currverse}%
590   \_else \_for num \versefrom..\verseto \_do{%
591     \_wlog{xxxxx v:\.currbook/\.currchapnum:##1}\_trymakedest{v:\.currbook/\.currchapnum:##1}}%
592   \_fi
593   \_raise5pt\_hbox{\_unless\_ifnum\.currversenum=1 \_markfont\.currversetext\_fi}%
594   \_prebuff\.buff \_space
595 }
596 \_def\.printchap{\_bigskip}
597
598 \_def\.printbeforefirst{%
599   \_par\_nobreak \_medskip
600   \.printchapnote
601   \_setbox0=\_vtop{\_kern-1.5ex \_ewref\_sxdef{ch!\.currbook/\_the\chapnum}{\_string\mypage}}
602   \_hbox{\_setfontsize{at50pt}\_bf\LiRed\_the\chapnum}}
603   \_dp0=0pt
604   \_tmpdim=\_lrmargin
605   \_advance\_tmpdim by4pt
606   \_ifnum\_the\chapnum>9 \_advance\_tmpdim by19pt \_fi
607   \_ifodd\_trycs{ch!\.currbook/\_the\chapnum}{0}
608   \_moveright\_tmpdim \_line{\_hss\_box0}
609   \_else \_moveleft\_tmpdim \_box0 \_fi
610   \_nobreak \_vskip-\_medskipamount
611   \_nobreak \_nointerlineskip \_noindent
612 }
613 \_def\.printchapnote{%
614   \_ifcsname chapnote!\.currbook/\_the\chapnum:0\_endcsname
615     {\_leftskip=\_parindent plus1fill \_rightskip=\_leftskip
616      \_noindent\_it \_cs{chapnote!\.currbook/\_the\chapnum:0}\_par}
617   \_medskip
618   \_fi
619 }

```

## 8 Bible references

We prepare temporary macros first.

`\.isspacein`  $\langle text \rangle$  `\_iftrue` is true if  $\langle text \rangle$  includes a space.

`\.iscolonin`  $\langle text \rangle$  `\_iftrue` is true if  $\langle text \rangle$  includes a colon.

`\.isdivisin`  $\langle text \rangle$  `\_iftrue` is true if  $\langle text \rangle$  includes a divis.

op-bible.opm

```

631 \_def\.isspacein #1 #2\_iftrue{\_isempty{#2}\_iffalse}
632 \_def\.iscolonin #1:#2\_iftrue{\_isempty{#2}\_iffalse}
633 \_def\.isdivisin #1-#2\_iftrue{\_isempty{#2}\_iffalse}

```

The `<` will be set to active as character equivalent to the macro `\.bref` $\langle text \rangle$ . This macro does all job with the hyperlinks. First of all, it scans the parts of the  $\langle text \rangle$  and saves them to

- `\.ltextP` ... the text before a link specification (given in "...")
- `\.ltextB` ... the book mark followed by ~
- `\.ltextC` ... the chapter number followed by :
- `\.ltextV` ... the verse number
- `\.ltextS` ... sub-verse identifier (a if there is a verse 4a)



- `\.ltextF` ... the -- if the  $\langle from \rangle - \langle to \rangle$  format is given
- `\.ltextN` ... the  $\langle to \rangle$  part from the  $\langle from \rangle - \langle to \rangle$  format.

All these macros above can be empty if the appropriate part of the scanned  $\langle text \rangle$  is missing. The `\.linkpre` macro includes `v` if it is verse link, includes `n` if it is note link and `g` if it is gloss link. These macros will be converted due to `\renum` data (if needed) and printed by `\.linktext`.

op-bible.opm

```

656 \_def\.linktext{\.ltextP\.ltextB\.ltextC\.ltextV\.ltextS\.ltextF\.ltextN}
657 \_def\.bref #1>{\_let\.brefH=\_relax \_def\.linkspec{#1}\_isnextchar{\.brefA}{\.brefA""}\_fi}
658 \_def\.brefA"#1"{\_def\.ltextP{#1}%
659   \_isnextchar{ }{\_addto\.ltextP{~}\_afterassignment\.brefB\_let\.next= }%
660   {\_isnextchar{\_def\.brefH{\_afterassignment\.brefB\_let\.next= }\.brefB}}%
661 }
662 \_def\.brefB #1>{\_fi #1 is link-spec
663   \_def\.ltextB{\_def\.ltextC{\_def\.ltextF{\_def\.ltextN{\_fi
664     \_isspacein #1 \_iftrue
665       \_iscolonin #1:\_iftrue \.brefBookChapterVerse #1>%
666       \_else \.brefBookChapter #1>\_fi
667     \_else \_iscolonin #1:\_iftrue \.brefChapterVerse #1>%
668     \_else \.brefVerse #1>%
669     \_fi\_fi
670     \_def\.linkpre{v}%
671     \_isnextchar n{\_def\.linkpre{n}\.brefC}%
672     {\_isnextchar g{\_def\.linkpre{g}\.brefC}%
673     {\_isnextchar a{\_def\.linkpre{a}\.brefC}%
674     {\_isnextchar i{\_def\.linkpre{i}\.brefC}{\.brefD}}}%
675 }
676 \_def\.brefC{\_afterassignment\.brefD \_let\.next= }
677
678 \_def\.brefBookChapterVerse #1 #2:#3>{\_def\.ltextB{#1~}\.brefChapterVerse #2:#3>}
679 \_def\.brefBookChapter #1 #2>{\_def\.ltextB{#1~}%
680   \_isinlist\nochapbooks{ #1 }\_iftrue
681     \_def\.ltextC{\_let\.ltextCin=\.ltextnCin \_afterfi{\.brefVerse #2>}%
682     \_else \_afterfi{\.brefChapter #2>}\_fi}
683 \_def\.brefChapterVerse #1:#2>{\_def\.ltextC{#1:}\.brefVerse #2>}
684 \_def\.brefVerse #1>{\_fi
685   \_isdivisin #1-\_iftrue \.brefFromTo #1>%
686   \_else \.versedef#1\_relax\_fi
687 }
688 \_def\.brefChapter #1>{\_fi
689   \_isdivisin #1-\_iftrue \.brefFromTo #1>\_let\.ltextC=\.ltextV
690   \_else \_def\.ltextC{#1}\_fi
691   \_def\.ltextV{\_def\.ltextS{\_fi
692 }
693 \_def\.brefFromTo #1-#2>{\.versedef#1\_relax\_def\.ltextF{--}\_def\.ltextN{#2}}

```

Because the verse number can be in the format 11b, we need to separate the numeric part of this and save it to `\.ltextV` and the rest is saved to `\.ltextS`. This is done by the `\.versedef  $\langle verse \rangle$  \_relax` macro.

op-bible.opm

```

701 \_def\.versedef {\_afterassignment\.versedefB \_tmpnum=0}
702 \_def\.versedefB #1\_relax{\_edef\.ltextV{\_the\_tmpnum}\_def\.ltextS{#1}}

```

Now, we create `\.linkfspec` from scanned data. It is  $\langle full-vref \rangle$  used for hyperlinks. We must manage all situations of incomplete links.

op-bible.opm

```

709 \_def\.brefD{%
710   \_ifnum 0\.ltextV=0 \_def\.ltextV{\_fi
711   \_if a\.linkpre \_ifx\.ltextV\_empty \_else \_edef\.ltextC{\.ltextV:\_def\.ltextV{\_fi\_fi
712   \_edef\.linkfspec{\_ea\.ltextBin\.ltextB-/\_ea\.ltextCin\.ltextC:/\_ea\.ltextVin\.ltextV:/}%
713   \.brefL
714 }
715 \_def\.ltextBin #1-#2/{\_ifx^#1~\.prelinkB \_else #1\_immediateassignment\_def\.prelinkB{#1}\_fi}/
716 \_def\.ltextCin #1:#2/{\_ifx^#1~\.prelinkC \_else #1\_immediateassignment\_def\.prelinkC{#1}\_fi:}
717 \_def\.ltextVin #1:#2/{\_ifx^#1~\.prelinkV \_else #1\_immediateassignment\_def\.prelinkV{#1}\_fi}
718 \_def\.ltextnCin #1:#2/{\_prelinkC:\_immediateassignment\_let\.ltextCin=\.ltextnCin}
719 \_let\.ltextScin=\.ltextCin

```

`\.prelinkB` is  $\langle book-mark \rangle$  of last referenced book. `\.prelinkC` is  $\langle chapter-num \rangle$  of last referenced chapter. They are used if the reference is not full. They are initialized at the beginning of books and chapters and they are changed locally in the `\Note` text. If the `<` is used then they are re-initialized.

```
729 \_def<\_let\.prelinkB=\.currbook \_let\.prelinkC=\.currchapnum \_let\.prelinkV=\.currversenum \.brief}
```

`\.oncebref` includes an additional macros which have to be processed in the single link, for example `\reduceref`. The `\everybref` token list includes macros which have to be applied for all links.

```
737 \_newtoks\everybref
738 \_def\.oncebref{}
739 \_nspublic \everybref ;
```

Macro `\.brefL` recalculates `\.linkfspec` and `\.linktext` due to `\renum` data and creates the link `\.linkpre:\.linkfspec` with the text `\.linktext`.

`\.renumlinktext  $\langle full-vref-ori \rangle$  \_relax  $\langle full-vref-modified \rangle$  \_relax` does re-calculation of the parts of the `\.linktext` macro.

The `\.linkfspecone` solves situation when chapter is given but no verse number: we must set the verse number to 1.

If the link destination is article, then the  $\langle full-vref \rangle$  has reduced format  $\langle book \rangle / \langle chapter \rangle$ . If the link destination is introduction then the  $\langle full-vref \rangle$  has more reduced format:  $\langle book \rangle /$ .

`\.linklog  $\{ \langle text \rangle \}$`  macro prints logging info of the link in the format

$\langle \langle link-spec \rangle \rangle = [ \langle full-vref \rangle ] \{ \langle printed-link \rangle \}$

`\.linklog` is `\wlog` by default and when `\tracinglinks` is set. It is `\ignreit` when `\notracinglinks` is set. You can set it to `\wterm` if you want.

```
760 \_def\.brefL{%
761   \_edef\.linkfspecm{\_ea\.renumvref\.linkfspec\_relax}%
762   \_ifx\.linkfspec\.linkfspecm \_else
763     \_ea\_ea\_ea\.renumlinktext \_ea\.linkfspec \_ea\_relax \.linkfspecm \_relax
764     \_let\.linkfspec=\.linkfspecm
765   \_fi
766   \_ifx\.ltextV\_empty \_ifx\.ltextC\_empty \_else \_ea\.linkfspecone \.linkfspec\_end \_fi\_fi
767   \_if a\.linkpre\_relax \_ea\.linkfspecarticle \.linkfspec\_end \_fi
768   \_if i\.linkpre\_relax \_ea\.linkfspecintro \.linkfspec\_end \_fi
769   \.linklog{\_sspace <\.linkspec>\.linkpost = [\linkpre:\linkspec]%
770     {\_ifx\.brefH\_empty\_unexpanded\_ea{\.ltextP}\_else \.linktext\_fi}}%
771   \.ensuredest \.createlink
772 }
773 \_def\.linkfspecone #1/#2\_end {\_def\.linkspec{#1:1}\_def\.prelinkV{1}}
774 \_def\.linkfspecarticle #1/#2:#3\_end {\_def\.linkspec{#1/#2}}
775 \_def\.linkfspecintro #1/#2\_end {\_def\.linkspec{#1/}}
776
777 \_def\.renumlinktext #1/#2:#3\_relax #4/#5:#6\_relax{%
778   \_ifx\.ltextC\_empty \_else \_def\.ltextC{#5}\_fi
779   \_def\.ltextV{#6}%
780   \_ifx\.ltextN\_empty \_else
781     \_ifx\.ltextF\.ltextDD
782       \_isinlist\.ltextN{:}\_iftrue
783       \_ifcsname rn!\tmark!#1/\.ltextN\_endcsname \_edef\.ltextN{\_cs{rn!\tmark!#1/\.ltextN}}\_fi
784       \_else \_edef\.ltextN{\_the\_numexpr#6+\.ltextN-#3\_relax}\_fi
785       \_else \_let\.tmp=\_ignoreit % \.ltextN is a list of verses, for example 7,9,13
786       \_ea\_foreach\.ltextN,\_do #1,{\_edef\.tmp{\_ea\_the\_numexpr#6+##1-#3}}%
787       \_let\.ltextN=\.tmp
788     \_fi
789   \_fi
790 }
791 \_def\.ltextDD{--}
792
793 \_def\.sspace{\_space\_space\_space\_space}
794 \_def\.linkpost{\_if v\.linkpre \_else \.linkpre\_fi \_space}
```

`\tracinglinks` and `\notracinglinks` are defined here.

```
800 \_def\tracinglinks{\_let\.linklog=\_wlog}
801 \_def\notracinglinks{\_let\.linklog=\_ignoreit}
802 \tracinglinks
```

`\.createlink` creates link only if it refers to the place of printed book because we don't want to see many warnings about unreferenced links when we try to print only selected books. It creates link `\.linkpre:\.linkfspec` with the text `\.linktext`

The link is created only if the book is to be printed, i.e. the `\pbook!⟨book⟩` is defined. The link is created always if a user declared `\tracingallrefs`.

op-bible.opm

```
814 \_def\_.createlink{%
815   \_ifx\_.briefH\_empty \_let\_.linktext=\.ltextP\_fi
816   \_ea\_.isprintedbook\_.linkfspec \_iftrue
817   \_link[\.linkpre:\.linkfspec]{\Blue}{\_.linktext}%
818   \_else {\Blue\_.linktext}\_fi}%
819 }
820 \_def\_.isprintedbook #1/#2\_iftrue{\_ifcsname pbook!#1\_endcsname}
821 \_def\tracingouterlinks{\_def\_.isprintedbook ##1\_iftrue{\_iftrue}}
```

We don't create destinations for all verses, notes etc. but only for those which are referenced. The macro `\.ensuredest` is called from `\.createlink` and it saves immediately `\sdef{⟨link⟩:⟨full-vref⟩}{}` to the special file `\jobname.xrf`. And the macro `\pg` saves immediately `\sdef{pg:⟨link⟩:⟨full-vref⟩}{??}` to this file. This `.xrf` file is read before standard `.ref` file. All link destinations save `\.Xdest{⟨full-vref⟩}` to the `.ref` file. The macro `\.Xdest` does nothing if `\pg:⟨link⟩:⟨full-vref⟩` is not defined (from `.rfx` file). Otherwise, it is defined as a correct pageno. This result is used in the `\pg` macro. If `\⟨link⟩:⟨full-vref⟩` is not defined, no link destination is created. First `TeX` run creates `.ref` and `.xrf` files and does not create any hyperlink destinations. Second `TeX` run uses data from these files and create correct hyperlinks and page numbers.

op-bible.opm

```
841 \_newwrite\_.xrf
842 \_immediate\_openout\_.xrf=\_jobname.xrf
843 \_openref
844
845 \_def\_.ensuredest{\_immediate\_write\_.xrf{\_string\_sdef{\.linkpre:\.linkfspec}{}}}
846 \_refdecl{
847   \_isfile{\_jobname.xrf}\_iftrue \_input{\_jobname.xrf}\_fi^^J
848   \_def\_.Xdest#1{\_ifcsname pg:#1\_endcsname \_sxdef{pg:#1}{\_ea\_usesecond\_currrpage}\_fi}^^J
849   \_def\_.mypage{\_ea\_usesecond\_currrpage}
850 }
851 \_def\_.trymakedest#1{%
852   \_ifcsname #1\_endcsname \_dest[#1]\_ea\_glet\_csname #1\_endcsname \_undefined \_fi
853   \_ewref\_.Xdest{#1}}%
854 }
```

The `\pg` macro should be used after `<...>`, i.e. the `\.linkpre` and `\.linkfspec` are defined. We use them. And the page number is saved to the `\pg:⟨link⟩:⟨full-vref⟩` macro in the second `TeX` run.

op-bible.opm

```
862 \_def\_.pg{%
863   \_ifcsname pg:\.linkpre:\.linkfspec\_endcsname
864     {\_edef\_.linktext{\_cs{pg:\.linkpre:\.linkfspec}}\_let\_.briefH=\_relax \.createlink}%
865   \_else {\Red ??}\_fi
866   \_immediate\_write\_.xrf{\_string\_sdef{pg:\.linkpre:\.linkfspec}{??}}%
867 }
868 \_nspublic \pg ;
```

## 9 Language variants

`\variants ⟨number-of-variants⟩ {⟨tmark-A⟩} {⟨tmark-B⟩} {⟨tmark-C⟩} ...`

sets `\.numvariants=⟨number-of-variants⟩` and does `\def\tmarkA{⟨tmark-A⟩} \def\var!1{⟨tmarkA⟩} \def\var!2{⟨tmark-B⟩} \def\var!3{⟨tmark-C⟩}` etc.

op-bible.opm

```
880 \_newcount\_.numvariants
881 \_def\_.variants{\_tmpnum=0 \_afterassignment\_.variantsA \_.numvariants}
882 \_def\_.variantsA{%
883   \_ifnum\_tmpnum<\_.numvariants
884     \_advance\_tmpnum by1
885     \_afterfi{\_.variantsB{\_the\_tmpnum}}%
886   \_fi
887 }
```

```

888 \_def\.\variantsB#1#2{%
889 \_ifnum#1=1 \_gdef\tmarkA{#2}\_sxdef{var!1}{#2}%
890 \_else \_sxdef{var!#1}{#2}%
891 \_fi
892 \.variantsA
893 }
894 \_nspublic \variants ;

```

`\vdef {⟨phrase-A⟩} {⟨phrase-B⟩} {⟨phrase-C⟩} ...` does

`\def\v!⟨tmark-B⟩!⟨phrase-A⟩{⟨phrase-B⟩} \def\v!⟨tmark-C⟩!⟨phrase-A⟩{⟨phrase-C⟩}` etc. Empty parameter is interpreted as undefined data. The internal macro `\.vdefB` implements the error message if there is too few parameters of `\.vdef` and we were read next `\.vdef`. The `\.sedef` used in the `\.vdefB{⟨number⟩}{⟨param⟩}` does real work and it defines (roughly sepaking):

```

If ⟨param⟩ is " \def \v!⟨tmark⟩!⟨phrase-A⟩ {⟨previous param⟩}
else \def \v!⟨tmark⟩!⟨phrase-A⟩ {⟨param⟩}

```

op-bible.opm

```

911 \_def\.\vdef#1{\_def\.\tmp{#1}%
912 \_ifcsname v!\_trycs{var!2}{!}\.\tmp\_endcsname
913 \_printwarn{\_noexpand\vdef used secondly for phrase {\.\tmp}, ignored}\_fi
914 \_tmpnum=1 \_ea\.\vdefA
915 }
916 \_def\.\vdefA{%
917 \_ifnum\_tmpnum<\.numvariants
918 \_advance\_tmpnum by1
919 \_afterfi{\.\vdefB{\_the\_tmpnum}}%
920 \_fi
921 }
922 \_def\.\vdefB#1#2{\_def\.\tmpa{#2}%
923 \_ifx\.\vdef#2\_def\.\tmpa{#2}\_fi
924 \_ifx\.\tmpa\_empty
925 \_ifx^#2^\_else
926 \_unless \_ifcsname v!\_cs{var!#1}!\.\tmp\_endcsname
927 \_sedef{v!\_cs{var!#1}!\.\tmp}{\_ifx"#2\.\prevcs{#1}\.\tmp \_else#2\_fi}%
928 \_fi\_fi
929 \_ea\.\vdefA
930 \_else \_errmessage{\_string\vdef: too few parameters. To be read again: \_string#2}%
931 \_ea\.\tmpa
932 \_fi
933 }
934 \_def\.\prevcs #1#2{\_ifnum#1=2 #2\_else \_cs{v!\_cs{var!\_the\_numexpr#1-1\_relax}!#2}\_fi}
935
936 \_nspublic \vdef ;

```

`\x/⟨phrase⟩/` expands to `\v!⟨tmark⟩!⟨phrase⟩` if such control sequence is defined else it expands simply to `⟨phrase⟩` using `\xA`. The `⟨tmark⟩` is actual value of the `\tmark` macro.

Note that if `\tmark` expands to `⟨t-markA⟩` (used in the `\variants` macro), then the `\v!⟨tmark⟩!⟨phrase⟩` is not defined and the `\x` macro expands to the `⟨phrase⟩` directly.

`\xA ⟨phrase⟩/` expands to `⟨phrase⟩` and prints warning, if `\tmark` is not the first `⟨t-markA⟩`.

op-bible.opm

```

949 \_def\.\x/#1/{\_trycs{v!\tmark!#1}{\_xA#1/}}
950 \_def\.\xA#1/{#1\_ifx\tmarkA\_undefined \_else \_ifx\tmark\tmarkA \_else
951 \_printwarn{\_string\x/#1/ -- this phrase is undefined by \_csstring\vdef}%
952 \_fi\_fi
953 }
954 \_nspublic \x ;

```

`\ww {⟨phrase-A⟩} {⟨phrase-B⟩} ...` has the same number of parameters as `\vdef`. They are separated by spaces. Each parameter can be in the “single form”, i.e. `{⟨phrase-A⟩}` or in the “extended form”, i.e. `{⟨phrase-A⟩}=⟨{printed-A}⟩`. The macro searches the correct phrase (given by the `\.varnum`) and saves it to the `\nextww`. The `\nextwwA` is set to `\nextww` if there is single form of the parameter else `\nextwwA` is `⟨printed-A⟩` part of the parameter in the extended form. These macros are used in the next `\Note` where they are re-set to `\undefined` meaning.

op-bible.opm

```

967 \_outer\_def\.\ww{%
968 \_ifx\.\varnum\_undefined \.setvarnum \_fi
969 \_tmpnum=0

```

```

970 \_ifx\.\nextww\undefined \_ea\wwA
971 \_else \.printwarn{Only single \_csstring\ww must be before \_csstring\Note}%
972 \_ea\wwB \_fi
973 }
974 \_def\wwA#1#2 {\_advance\_tmpnum by1
975 \_def\.\nextww{#1}\_def\.\nextwwA{#2}%
976 \_ifx\.\nextwwA\_empty \_let\.\nextwwA=\.\nextww \_else \_ea \.redefwwA #2\_end \_fi
977 \_ifnum\.\varnum=\_tmpnum \_ifnum\_tmpnum<\.numvariants \_ea\_ea\_ea \.wwB \_fi
978 \_else \_ea \.wwA \_fi
979 }
980 \_def\wwB#1 {\_advance\_tmpnum by1
981 \_ifnum\_tmpnum<\.numvariants \_ea\wwB \_fi
982 }
983 \_def\.\redefwwA =#1\_end{\_def\.\nextwwA{#1}}
984
985 \_nspublic \ww ;

```

The `\switch` macro reads a pair of parameters using `\switchA` and processes the list of variants in `\foreach` loop. If an element from the list is equal with `\tmark` then the #2 (saved in `\switchD` token list) is run and next parameter pairs are read by `\switchN`, i.e. they are ignored.

The `\Note` and `\ww` are defined as `\outer` in order to better diagnose mistakes with number of parameters of `\ww` or missig empty line after `\Note` text. But we want to skip such objects in `\switch` parameters. This is the reason why we run `\unsetouter` before the `\switch` parameter is read and we run `\setouter` in order to return to the normal setting.

op-bible.opm

```

1000 \_newtoks\switchD
1001 \_def\.\switch {\_let\.\switchN=\.\switchA \.unsetouter \.switchN}
1002 \_long\_def\.\switchA #1#2{\.switchD={\.setouter #2\_let\.\switchN=\.switchI}%
1003 \_ifx\_relax#1\_relax \_the\switchD
1004 \_else \_foreach #1,\_do ##1,{\_def\tmp{##1}\.switchC}%
1005 \_fi
1006 \_futurelet\.\next\switchB
1007 }
1008 \_def\.\switchB{\_ifx\.\next\_bgroup \.unsetouter \_ea\.\switchN \_else \.setouter \_fi}
1009 \_long\_def\.\switchI #1#2{\_futurelet\.\next\switchB}
1010 \_def\.\switchC{\_ifx\tmp\tmark \_the\switchD \_fi}
1011 \_def\.\unsetouter{\_slet{ww}{\_relax}\_slet{Note}{\_relax}}
1012 \_def\.\setouter{\_slet{ww}{\_opb\_iww}\_slet{Note}{\_opb\_iNote}}
1013 \_let\.\iww=\.\ww % backup of outer ww
1014 \_let\.\iNote=\.\Note % backup of outer Note
1015
1016 \_nspublic \switch ;

```

`\setvarnum` sets the `\varnum` as the position number of the current language variant due to the value of `\tmark`. The `\variants` declaration must precede.

op-bible.opm

```

1024 \_def\.\setvarnum{\_gdef\.\varnum{0}%
1025 \_ifnum\.\numvariants=0 \_gdef\.\varnum{1}\_wlog{There is only single language variant (1)}%
1026 \_else
1027 \_tmpnum=0
1028 \_loop
1029 \_advance\_tmpnum by1
1030 \_ea\_ifx \_csname var!\_the\_tmpnum\_endcsname \tmark \_xdef\.\varnum{\_the\_tmpnum}\_fi
1031 \_ifnum\_tmpnum<\.numvariants \_repeat
1032 \_ifnum \.\varnum=0 \_errmessage{\_noexpand\tmark isn't set, \_noexpand\.\setvarnum failed}%
1033 \_else \_wlog{Language variant set by \_string\tmark\tmark (\.\varnum)}\_fi
1034 \_fi
1035 }

```

`\renum <book-mark> <chapter-num>:<verse-num> = <t-mark> <chap-num>:<from>-<to>` does

```

\def \rn!<t-mark>!<full-vref>{<chap-num>:<from>}
\def \rn!<t-mark>!<full-vref+1>{<chap-num>:<from+1>}
\def \rn!<t-mark>!<full-vref+2>{<chap-num>:<from+2>}
... etc.
\def \rn!<t-mark>!<full-vref+n>{<chap-num>:<to>}

```

```

1049 \_def\renum #1 #2:#3 = #4 #5:#6-#7 {%
1050   \_tmpnum=#3\_relax
1051   \_forum #6..#7 \_do {\_sxdef{rn!#4!#1/#2:\_the\_tmpnum}{#5:#1}\_incr\_tmpnum}%
1052 }
1053 \_nspublic \renum ;

```

## 10 Inserting notes to the page

We declare new insert `\.noteins` used in the `\output` routine.

op-bible.opm

```

1062 \_newinsert \.noteins
1063 \_skip\.noteins=\_bigskipamount % noterule height
1064 \_count\.noteins=500 % two columns
1065 \_dimen\.noteins=\_maxdimen % full page of notes allowed

```

The `\.noteinsert {<text>}` inserts its parameter to the `\.noteins`. We open the `\insert` and set basic parameters using `\.noteset`. Then the empty box with strut height is inserted in vertical mode (in order to consecutive notes have good baselineskip between them). Then the `<text>` is printed and the paragraph is finalized. The empty box with strut depth is appended after the paragraph (in order to the same reason). Final `\penalty0` allows breaking between notes.

op-bible.opm

```

1078 \_def\.noteinsert #1{\_insert\.noteins{%
1079   \.noteset
1080   \_vbox to\_ht\_strutbox{\\_nobreak \_vskip-\_baselineskip
1081   #1\_unskip\_par \_nobreak \_vskip-\_baselineskip
1082   \_hbox{\_lower\_dp\_strutbox\_vbox{}}
1083   \_penalty0
1084 }}
1085 \_def\.noteset{\Heros\cond \_scalemain \_typoscale[800/800] % Heros condensed 80%
1086   \Black \_nobreak
1087   \_widowpenalty=20 \_clubpenalty=20
1088   \_leftskip=0pt \_rightskip=0pt \_parfillskip=0pt plus1fill
1089   \_parindent=0pt
1090   \_lineskiplimit=-3pt
1091   \_hsize=.5\_hsize \_advance\_hsize by-1em\_relax % two columns
1092   \_everypar{}
1093 }

```

We add macros for inserting two columns of notes from `\.noteins` into the page. First, we add `\noterule` with the space given by `\skip\.noteins`. The `\.noteins` material is prefixed by `\penalty0` (in order to allow the next `\vsplit` operation) and the `\vfil` is added (in order to the case when the second column is smaller than the first one). The `\splittopskip` is set and first `\vsplit to0pt` adds skip given by `\splittopskip` to the `\.noteins`. The `\_balancecolumns` from OpTeX for splitting to two columns is used. We need to set `\_Ncols`, `\_dimen0` and `\_box6` before running `\_balancecolumns`. We need to insert `\vskip\splittopskip` because `\_balancecolumns` supposes that the typesetting point resides at the first baseline of the columns.

The final `\vskip` does “raggedbottom”. We need to add `1filll` in order to suppress the `\vfill` from the `\end` algorithm. We add `minus6pt` because the height of two columns can be by half-line higher than the insertion algorithm expects (in the case with odd lines before splitting to the two columns).

op-bible.opm

```

1114 \_addto\_pagecontents{%
1115   \_ifvoid\.noteins \_else
1116     \_vskip\_skip\.noteins \noterule
1117     \_setbox\.noteins=\_vbox{\_penalty0 \_unvbox\.noteins \_vfil}
1118     \_splittopskip=12pt
1119     \_setbox0=\_vsplit\.noteins to0pt % adding \splittopskip to \.noteins
1120     \_def\_Ncols{2}
1121     \_dimen0=.5\_ht\.noteins \_setbox6=\_box\.noteins
1122     \_vskip\_splittopskip
1123     \_balancecolumns
1124     \_fi
1125     \_unless\_ifvoid\.botins \_unvbox\.botins
1126     \_else \_vskip 0pt plus1filll minus8pt \_fi
1127 }
1128 \_def \noterule {\_kern-3pt {\Black \_hrule width\_hsize}\_kern 2.6pt }

```

## 11 Inserting images and articles to the page

`\.botins` is analogue insert as `\_topins` but the material is inserted to the bottom of the page. The material is created by `\.botinsert...\.endbot` pair of control sequences. We use it for inserting images and articles to the page.

op-bible.opm

```
1140 \_newinsert\.botins
1141 \_def\.botinsert{\_setbox0=\_vbox\_bgroup}
1142 \_def\.endbot{\_par\_egroup
1143   \_insert\.botins{\_splittopskip=0pt \_penalty100
1144     \_hrule height0pt \_nobreak\_medskip\_bigskip \_unvbox0
1145   }%
1146 }
1147 \_skip\.botins=\_zoskip    % no space added when a topinsert is present
1148 \_count\.botins=1000     % magnification factor (1 to 1)
1149 \_dimen\.botins=\_maxdimen % no limit per page
```

`\putImage <chapter>:<verse> {<title>} [<label>] (<params>) {<image-file>}` inserts the given image to the page where the beginning of the verse given by `<chapter>:<verse>` exists. We register a new action by `\.newaction{<full-vref>}{\doImage{<title>} [<label>] (<params>){<image-file>}}`. The `\doImage` puts the image by `\.botinsert...\.endbot` pair. The `\botTitle{<title>} [<label>]` prints the title of the image (or article or whatever is put to the bottom of the page) and inserts the destination of hyperlink based on the `<label>`, if the `<label>` isn't empty.

op-bible.opm

```
1162 \_def\.putImage #1 #2#3[#4]#5(#6)#7{% chap:verse {Title} [label] (params) {image-file.pdf}
1163   \_edef\.fullvref{\.gentovref{#1}}%
1164   \_edef\.fullvrefm{\_ea\.renumvref\.fullvref\_relax}%
1165   \_ea\.newaction\_ea{\.fullvrefm}{\doImage{#2}[#4] (#6){#7}}%
1166 }
1167 \_def\.doImage #1[#2] (#3)#4{% {Title}[label] (params){image-file.pdf}
1168   \.botinsert
1169     \.botTitle{#1}[#2]%
1170     \_kern3pt \_nobreak
1171     \_hbox{\picw=\hsize #3\inspic{#4}}%
1172   \.endbot
1173 }
1174 \_def\.botTitle#1[#2]{\_hbox{\_captionfont
1175   \_ifx^#2\_else \.botDest{#1}[#2]\_fi
1176   \_rlap{\Grey \_vrule height1.2em depth.5em width\_hsize}\White\_kern12pt #1}%
1177 }
1178 \_picdir={images/}
1179 \_def\.botDest#1[#2]{\_label{#2}\_wlabel{#1}}
1180
1181 \_nspublic \putImage ;
```

`\putArticle <chapter>:<verse> {<title>} [<label>] (<params>)` inserts an article given in the file `articles-*.tex` signed by `\Article [<label>]`. The article starts at the page where `<chapter>:<verse>` is or at the next page. The article is in two-columns style and it is divided to  $k$  two-columns parts each of them is inserted at the bottom of the next page.

We calculate the number of pages used for article text by following rules. All the two-columns parts have the same height. If there are more than one such a part, the height does not exceeds  $2/3$  of the page. But single two-column part can be higher.

`\putArticle` registers `\doArticle` using `\.newaction`. `\doArticle` is run at the beginning of given verse and creates an `\botisert`. The insert material is breakable at its beginig and between each two-column boxes created by the `\balancecolumn` macro.

We register a new action by `\.newaction{<full-vref>}{\doArticle{<title>} [<label>] (<params>)}`.

op-bible.opm

```
1205 \_newcount\.articlenum
1206 \_def\.putArticle #1 #2#3[#4]#5(#6){% chap:verse {Title} [number] (params)
1207   \_edef\.fullvref{\.gentovref{#1}}%
1208   \_edef\.fullvrefm{\_ea\.renumvref\.fullvref\_relax}%
1209   \_ea\.newaction\_ea{\.fullvrefm}{\doArticle{#2}[#4] (#6)}%
1210 }
1211 \_nspublic \putArticle ;
```

The `\doArticle {<Title>} [<label>] (<params>)` inserts the article to one or more pages by `\.botinsert...\.endbot`. The Article is printed to two columns per page, all collumns of the article is completely balanced. First,



the whole text is saved to the `\box0` with given column size and the number of pages is calculated in `\_tmpnum`. Then the number of columns `\_Ncols` is 2 times the number of calculated pages. The height of each two-columns part of the article is `\dimen0`. Finally we do re-boxing the output of `\_balancecolumns` in order to reach individual columns and create pairs of them by `\for` loop. These pairs are completed to blocks with LightGrey background. These blocks divided by `\break` are inserted into `\.botinsert`.

op-bible.opm

```

1227 \_def\doArticle#1[#2](#3){% {Title}[number] (params)
1228   \_incr\articlenum
1229   \.botinsert
1230   \_def\botDest##1[##2]{\trymakedest{a:\currbook/##2}}
1231   \_parindent=12pt \_iindent=\_parindent
1232   \_setbox0=\_vbox{\_hsize=.458\_hsize \_emergencystretch=1em
1233     \_hbadness=6000 \_baselineskip=\_dimexpr\_baselineskip plus1pt
1234     \_def\Article[##1]{\_endinput}
1235     \_penalty0
1236     \_long\_def\searcharticle##1\Article[#2]}
1237     \_ea\searcharticle \_input \articlefile \_relax}
1238   \_splittopskip=12pt
1239   \_setbox1=\_vsplit0 to0pt % adding \splittopskip
1240   \_tmpdim=\_vsize \_advance\_tmpdim by-24pt % \.botTitle height plus above/below skips
1241   \_ifdim 2\_tmpdim > \_ht0 \_tmpnum=1
1242   \_else
1243     \_tmpnum=\_roundexpr{\_bp{\_ht0}/\_bp{1.333\_vsize}+0.999} % number of 2/3 pages
1244   \_fi
1245   \_multiply\_tmpnum by2 % number of columns
1246   \_edef\_Ncols{\_the\_tmpnum}
1247   \_dimen0=\_expr{1/\_Ncols}\_ht0 \_setbox6=\_box0 % height of each two-columns part
1248   \_setbox0=\_vbox{\_balancecolumns}
1249   \_tmpdim=\_ht0 \_advance\_tmpdim by1.2\_baselineskip
1250   \_setbox0=\_vbox{\_unvbox0 \_global\_setbox2=\_lastbox}
1251   \_setbox0=\_hbox{\_unhbox2
1252     \_for num 1..\_Ncols \_do {\_unskip \_global\_setbox1##1=\_lastbox}}
1253     \_for numstep -2: \_Ncols..1 \_do {
1254       \_hrule height0pt\_kern5pt\_nobreak\_vfill
1255       \_ifnum\_Ncols=##1 \.botTitle{#1}[#2]\_else \.botTitle{}[]\_fi
1256       \_kern3pt \_nobreak
1257       \_hbox to\_hsize{%
1258         \_rlap{\_LightGrey \_vrule height\_tmpdim depth6pt width\_hsize}%
1259         \_kern\_parindent
1260         \_box1##1\_hss\_box1\_the\_numexpr##1-1
1261         \_kern\_parindent
1262       }
1263     }
1264   }
1265   \.endbot
1266 }
1267 \_def\roundexpr#1{\_ea\_ea\_ea\roundexprA\_expr{#1}\_relax}
1268 \_def\roundexprA#1.#2\_relax{\_ifnum#1=0 0\_else #1\_fi}

```

## 12 Inserting citations to the page

`\putCite <gen-vref> {<text>}` creates a citation `<text>` inserted to the top of the page where the verse `<gen-vref>` is. We register a new action by `\newaction{<full-vref>}{\dotopCite{<text>}}`.

op-bible.opm

```

1279 \_def\putCite #1 #2{% chap:verse {text}
1280   \_edef\fullvref{\gentovref{#1}}%
1281   \_edef\fullvrefm{\_ea\renumvref\fullvref\_relax}%
1282   \_ea\newaction\_ea\fullvrefm{\dotopCite{#2}}%
1283 }
1284 \_nspublic \putCite ;

```

`\dotopCite {<text>}` creates the citation text by `\topinsert... \endinsert` from plain TeX. We distinguish two cases: the citation on a left page and the citation on a right page. We sawe the page position using `\ewref` to the .ref file as `\sxddef{ct!<citenum>}{\_mypage}` and we know the page position in the second TeX run and use it in the `\ifodd` condition. The typesetting parameters differ in “left” and “right” case.

```

1296 \_newcount\.\citenum
1297 \_def\.\dotopCite #1{%
1298   \_topinsert
1299   \_typosize[12/16]\_bi
1300   \_incr\.\citenum
1301   \_ifodd \_trycs{ct!\_the\.\citenum}{0}\_relax
1302   \_leftskip=.3\_hsize plus1fil \_parfillskip=0pt
1303   \_noindent
1304   \_rlap{\_hskip\_hsize \_kern-\_leftskip \_copy\.\rqqbox}\_hfill
1305   \_else
1306   \_let\quotedby=\.\quotedbyright
1307   \_rightskip=.3\_hsize plus 1fil
1308   \_noindent \_llap{\_copy\.\lqqbox}%
1309   \_fi
1310   {\_printCite{#1}\_unskip}\_par
1311   \_ewref\_sxdef{{ct!\_the\.\citenum}{\_string\.\mypage}}}%
1312 % \_vskip-.3\baselineskip
1313 \_endinsert
1314 }
1315 \_def\.\printCite#1{\_pdfliteral{2 Tr .15 w .9 g}#1\_pdfliteral{0 Tr 0 w 0 g}}
1316 \_def\.\printCite#1{{\Grey#1}}

```

The `\.lqqbox` and `\.rqqbox` include the graphical marks for quotations. First one is used at the left pages, second one at the right pages.

The macro `\quotedby{<author>}` puts the author of the quotation to the next line. The macro `\quotedbyright` (which is used at left pages) prints the `<author>` at the last line if there is sufficient space.

```

1326 \_newbox\.\lqqbox
1327 \_newbox\.\rqqbox
1328 \_setbox\.\lqqbox=\_hbox{\_lower3pt\_hbox{\_setfontsize{at70pt}\_bf\LiRed,}}
1329 \_setbox\.\rqqbox=\_hbox{\_kern2pt\_lower38pt\_hbox{\_setfontsize{at70pt}\_bf\LiRed}}
1330 \_ht\.\lqqbox=0pt \_dp\.\lqqbox=0pt
1331 \_ht\.\rqqbox=0pt \_dp\.\rqqbox=0pt
1332
1333 \_def\quotedby{\_par}
1334 \_def\.\quotedbyright#1{%
1335   \_unskip\_nobreak\_hfill\_penalty0\_hskip2em
1336   \_null\_nobreak\_hskip\_iindent\_hbox{#1}}

```

The following macros `\Cite`, `\insertCite` and `\swapCites` are used for insertion of citations to the two-column printed articles. The `\Cite<label>{<text>}` simply saves the `<text>` to the macro `\c!<article-num>!<label>`. The `\insertCite<label>{<left-or-right>}` inserts the citation declared by `\Cite <label>` to the text using `\vadjust`. The variant `\left` and `\right` is processed or ignored. This depends on the parity of the current page, which is restored from `.ref` file and saved to the macro `\cp!<article-num>!<label>`.

```

1350 \_def\.\Cite #1#2{\_sdef{c!\_the\.\articlenum!#1}{#2}}
1351 \_def\.\insertCite #1#2{\_def\.\citelabel{#1}%
1352   \_ifx\_left#2\.\insertCiteleft
1353   \_else \_ifx#2\_right\.\insertCiteright\_else
1354   \_errmessage{\_noexpand\insertCite#1: \_noexpand\left or \_noexpand\right expected}%
1355   \_fi\_fi
1356 }
1357 \_def\.\insertCiteleft {%
1358   \_ifnum\.\citepg=1 \_printwarn{\_noexpand\.\insertCite\.\citelabel: \_noexpand\.\swapCites activated}\_fi
1359   \_ifodd \_numexpr\_trycs{cp!\_the\.\articlenum!\.\citelabel}{0}+\.\citepg\_relax
1360   \_else \.\insertCitelr \_left \_fi
1361 }
1362 \_def\.\insertCiteright{%
1363   \_ifodd \_numexpr\_trycs{cp!\_the\.\articlenum!\.\citelabel}{0}+\.\citepg\_relax
1364   \.\insertCitelr \_right \_fi
1365 }
1366 \_def\.\insertCitelr#1{\_unskip\_vadjust{\_vbox{%
1367   \_ewref\_sxdef{{cp!\_the\.\articlenum!\.\citelabel}{\_string\.\mypage}}}%
1368   \_vskip6pt
1369   \_advance\_hsize by\_parindent
1370   \_typosize[12/16]\_bi\Grey
1371   \_ifx#1\_left

```

```

1372 \_def\quotedby{\_par\_hfill}
1373 \_rightskip=\_parindent plus1fil \_leftskip=0pt
1374 \_setbox0\_vbox{%
1375 \_medskip \_noindent
1376 \_llap{\_copy\lqqbox}\_ignorespaces
1377 \_printCite{\_cs{c!\_the\articlenum!\.citelabel}}\_medskip}%
1378 \_hbox{\_kern-\_parindent\_rlap{White
1379 \_vrule height\_ht0 width\_hsize}\_box0}%
1380 \_else
1381 \_leftskip=\_parindent plus1fil
1382 \_parfillskip=0pt
1383 \_setbox0\_vbox{%
1384 \_medskip \_noindent
1385 \_rlap{\_hskip\_hsize\_kern-\_parindent\_copy\lqqbox}\_hfill
1386 \_ignorespaces \_printCite{\_cs{c!\_the\articlenum!\.citelabel}}\_medskip}%
1387 \_rlap{\_rlap{White \_vrule height\_ht0 width\_hsize}\_box0}%
1388 \_fi
1389 \_vskip6pt
1390 }}
1391 \_def\swapCites{\_def\citepg{1}}
1392 \_def\citepg{0}
1393
1394 \_nspublic \Cite \insertCite ;

```

## Insertions into the intro text

op-bible.opm

```

1402 %% TBN page 236
1403
1404 \_newcount\shapenum
1405 \_newdimen\ii \_newdimen\w
1406 \_def\oblom #1 od #2 odsadit #3 {\_par \.ii=#1 \.w=\_hsize
1407 \_ifdim\ii>\_zo \_advance\w by-\_ii
1408 \_else \_advance\w by\_ii \.ii=\_zo \_fi
1409 \shapenum=1 \_tmpnum=0 \_def\shapelist{}
1410 \_loop \_ifnum\shapenum<#2 \_edef\shapelist{\shapelist\_zo\_hsize}%
1411 \_advance\shapenum by1 \_repeat
1412 \_loop \_edef\shapelist{\shapelist\ii\w}%
1413 \_advance\_tmpnum by1 \_ifnum\_tmpnum<#3 \_repeat
1414 \_advance\shapenum by#3 \_edef\shapelist{\shapelist\_zo\_hsize}
1415 \doshape}
1416 \_def\doshape{\_parshape \shapenum \shapelist}
1417 \_newcount\globpar
1418 \_ifx\_partokenset \_undefined \_def\partoken{\par} \_else \_def\partoken{\_par} \_fi
1419 \_def\doshape{\_global\globpar=0 \_ea\_def\partoken{\_ifhmode\shapepar\_fi}}
1420 \_def\shapepar{\_prevgraf=\_globpar \_parshape\shapenum\shapelist
1421 \_endgraf \_global\globpar=\_prevgraf
1422 \_ifnum \_prevgraf>\shapenum \_ea\_let\partoken=\_endgraf \_fi
1423 }
1424
1425 \_def\Citehereleft #1 (#2) #3{{
1426 \_par
1427 \_def\quotedby{\_par\_hfill}
1428 \_rightskip=\_parindent plus1fil \_leftskip=0pt
1429 \_setbox0\_vbox{%
1430 \_typosize[12/16]\_bi\Grey
1431 \_hsize=.5\_hsize
1432 \_medskip \_noindent
1433 \_llap{\_copy\lqqbox}\_ignorespaces
1434 \_printCite{#3}\_medskip}}%
1435 \_tmpdim=\_ht0 \_advance\_tmpdim by\_baselineskip
1436 \_xdef\lines{\_the\_numexpr \_number\_tmpdim / \_number\_baselineskip \_relax}%
1437 \_nointerlineskip\_vbox to0pt{\_kern#1\_baselineskip #2
1438 \_hbox{\_rlap{White
1439 \_kern-3mm\_vrule height\_ht0 width.5\_hsize}\_box0}%
1440 \_vss}}
1441 \_tmpdim=\_hsize \_advance\_tmpdim by-2\_leftskip
1442 \oblom {5\_tmpdim} od #1 odsadit {\lines}
1443 }
1444 \_def\Citehereright #1 (#2) #3{{
1445 \_par

```

```

1446 \_def\quotedby{\_par\_parfillskip=0pt\_hfill}
1447 \_leftskip=\_parindent plusifill\_rightskip=0pt
1448 \_setbox0\_vbox{%
1449 \_typosize[12/16]\_bi\Grey
1450 \_hsize=.5\_hsize
1451 \_vskip\_medskipamount\_rlap{\_kern\_hsize\_copy\rqbox}\_vskip-\_medskipamount
1452 \_printCitef{\_noindent\_ignorespaces#3}\_medskip}}%
1453 \_tmpdim=\_ht0\_advance\_tmpdim by\_baselineskip
1454 \_xdef\_lines{\_the\_numexpr\_number\_tmpdim /\_number\_baselineskip\_relax}%
1455 \_nointerlineskip\_vbox toOpt{\_kern#1\_baselineskip #2
1456 \_hbox to\_hsize{\_hss
1457 \_llap{\White\_vrule height\_ht0 width.5\_hsize\_kern-3mm}%
1458 \_llap{\_box0}}
1459 \_vss}}
1460 \_tmpdim=\_hsize\_advance\_tmpdim by-2\_leftskip
1461 \_oblong {-0.5\_tmpdim} od #1 odsadit {\_lines}
1462 }
1463
1464 \_def\Citehere{\_par\_ifodd\_pageno\_ea\Citehereright\_else\_ea\Citehereleft\_fi}
1465
1466 \_nspublic\Citehere ;
1467
1468 \_def\insertBot #1#2[#3]#4(#5)#6{% {Title} [label] (params) {data}
1469 \_botinsert
1470 \_botTitle{#1}[#3]%
1471 \_kern3pt\_nobreak
1472 \_vbox{\_picwidth=\_hsize #5 #6}%
1473 \_endbot
1474 }
1475 \_def\putBot #1 #2#3[#4]#5(#6)#7{% chap:verse {Title} [label] (params) {image-file.pdf}
1476 \_edef\_fullvref{\_gentovref{#1}}%
1477 \_edef\_fullvrefm{\_ea\_renumvref\_fullvref\_relax}%
1478 \_ea\_newaction\_ea{\_fullvrefm}{\insertBot{#2}[#4](#6){#7}}%
1479 }
1480
1481 \_def\_c[#1/#2]#3{% text podel krivky: \c[init-rotace/repetice]{text}
1482 \_pdfsave\_pdfrotate{#1}\_rlap{\_edef\_tmpb{#3}\_replstring\_tmpb{ }{ } }\_def\_tmpa{#2}%
1483 \_ea\_foreach\_tmpb\_do{##1\_tmpa}}\_pdfrestore\_kern10mm
1484 }
1485 \_nspublic \insertBot \putBot \c ;

```

**\.printintro** macro (by default) prints the introduction of the book from the **\introfile**, prints the title "Introduction" (depending on the current language and puts all introduction text between **\.begblock** and **\.endblock**.

op-bible.opm

```

1494 \_def\.printintro{%
1495 \_begblock
1496 \_dest[i:\_currbook/]
1497 \_chaptit{\_mtext{intro}}}%
1498 \_input{\_introfile}
1499 \_endblock
1500 }

```

Text block with grey background splittable to more pages is between **\.begblock** and **\.endblock** macros. It is used for introduction text. See also OpTeX trick 0031.

op-bible.opm

```

1508 \_newcount\.blocklevel % nesting level of blocks
1509 \_def\.begblock{\_par\_bgroup
1510 \_advance\.blocklevel by1\_advance\_leftskip by\_iindent\_rightskip=\_leftskip
1511 \_medskip
1512 \_pdfsavepos\_ea\_wref\_ea\Xblock\_ea{\_ea{\_the\.blocklevel}B{\_the\_pdflasttypos}}
1513 \_nobreak\_medskip
1514 }
1515 \_def\.endblock{\_par\_nobreak\_medskip
1516 \_pdfsavepos\_ea\_wref\_ea\Xblock\_ea{\_ea{\_the\.blocklevel}E{\_the\_pdflasttypos}}
1517 \_medskip\_egroup
1518 }
1519 \_refdecl{%
1520 \_def\Xblock#1#2#3{\_ifnum#1=1\_edef\_tmp{frm:\_ea\_ignoresecond\_currpage}~^J

```

```

1521 \_unless\_ifcsname \_tmp \_endcsname \_sxdef{\_tmp}{\_fi^^J
1522 \_sxdef{\_tmp}{\_cs{\_tmp}#2{#3}}\_fi}
1523 }
1524 \_newdimen\_frtop \_newdimen\_frbottom % positions of top and bottom text on the pages
1525 \_def\_frcolor{.93 g } % light grey -- color of blocks.
1526 \_pgbackground={%
1527 \_slet{\_opb\_tmp}{frm:\_the\_gpageno}
1528 \_ifx{\_tmp}\_undefined \_def{\_tmp}{\_fi
1529 \_frtop=\_dimexpr \_pdfpageheight-\_voffset+\_smallskipamount\_relax
1530 \_frbottom=\_dimexpr \_pdfpageheight-\_voffset-\_vsize-\_medskipamount\_relax
1531 \_ifx{\_frnext y \_edef{\_tmp{B{\_number\_frtop}\_tmp}\_global\_let\_frnext n\_fi
1532 \_ea\_.printframes \_tmp B{0}E{\_number\_frbottom}
1533 \_ifx{\_frameslist\_empty \_else
1534 \_pdfliteral{q \_frcolor 1 0 0 1 0 \_bp{-\_pdfpageheight} cm \_frameslist Q}\_fi
1535 }
1536 \_def\_.printframes B#1#2E#3{\_ifnum#1=0 \_else
1537 \_.printframe {\_hoffset}{#3sp}{\_xhsize}{\_ifnum#1=-1 \_number\_frtop\_else#1\_fi sp-#3sp}
1538 \_ifx^#2\_else \_global\_let\_frnext=y \_let\_.printframes=\_relax \_fi
1539 \_ea\_.printframes\_fi
1540 }
1541 \_def\_.frameslist{}
1542 \_def\_.printframe #1#2#3#4{\_edef\_.frameslist{\_.frameslist
1543 \_bp{#1} \_bp{#2} \_bp{#3} \_bp{#4} re f }%
1544 }

```

## 13 Outline

op-bible.opm

```

1552 \_newdimen\_colsep
1553 \_colsep=10pt
1554
1555 \_def\_.Outline{
1556 \_medskip
1557 % \_filbreak
1558 \_chaptit{\_mtext{outline}}}%
1559 \_everylist={\_ifcase\_ilevel \_or \_style I \_or \_style A \_or \_style n \_fi}
1560 \_sdef{\_item:A}{\_strut\_uppercase\_ea{\_athe\_itemnum}. }
1561 \_sdef{\_item:I}{\_strut\_uppercase\_ea{\_romannumeral\_itemnum}. }
1562 \_hsize=.5\_hsize \_advance\_hsize by-\_colsep
1563 \_emergencystretch=40pt
1564 \_leftskip=0pt \_rightskip=0pt
1565 }
1566 \_def\_.rightnote#1{\_par
1567 \_setbox0=\_hbox{\_kern\_hsize \_kern\_.colsep
1568 \_vtop{\_leftskip=0pt \_kern0pt\_noindent\_strut\_it#1}}
1569 \_ht0=0pt \_dp0=0pt \_box0 \_nointerlineskip
1570 }
1571 \_nspublic \_Outline \_rightnote ;

```

## 14 Typesetting variants

By default, chapter numbers are in the outer margin and quotes characters too. The `\normalchapnums` macro moves chapter numbers to the left side in the first paragraph, cquotes characters are removed and outer margins are reduced because there is no material in them.

op-bible.opm

```

1585 \_def\_.normalchapnumbers{
1586 \_margins/2 a2 (25,25,20,20)mm
1587 \_lrmargin=0pt
1588 \_setbox0=\_box\_.lqqbox \_setbox0=\_box\_.rqqbox
1589 \_def\_.printbeforefirst{%
1590 \_nobreak\_medskip
1591 \_printchapnote
1592 \_hangindent=\_parindent \_hangafter=-2
1593 \_noindent \_llap{\_vbox to0pt
1594 {\_kern-8pt\_hbox{\_setfontsize{at23pt}\_bf\Red\_the\_.chapnum\_kern5pt}\_vss}}%
1595 }
1596 }
1597 \_nspublic \_normalchapnumbers ;

```

## 15 Checking syntax

op-bible.opm

```
1605 \_def\checksyntax#1 {%
1606   \_let\processbooks=\_relax
1607   \_ifx\_relax#1\_relax \_else
1608     \_begingroup
1609       \_the\syntaxmacros
1610       \_wterm{^^J** checking file: #1 **^^J}
1611       \_input{#1}
1612       \_vfil\_break
1613     \_endgroup
1614   \_ea\checksyntax \_fi
1615 }
1616
1617 \_newtoks\syntaxmacros
1618 {\_catcode\<=13
1619 \_global\syntaxmacros={
1620   \_def<#1>{\_bgroup
1621     \_message{checking \_unexpanded{<#1>}}%
1622     \_ifx\_relax#1\_relax \_errmessage{empty link}\_nobref\_else \_afterfi{\_checkbref#1>\_bref#1>}\_fi
1623     \_glet\linkpre=\_linkpre \_glet\linkfspec=\_linkfspec
1624   \_egroup
1625 }
1626 \_def\checkbref#1#2>{%
1627   \_isinlist{.#1#2}{<}\_iftrue \_errmessage{duplicated \_string<}\_nobref\_else
1628   \_ifx"#1\checkbrefQ #1#2>\_else \_checkbrefD #1#2>\_fi\_fi
1629 }
1630 \_def\checkbrefQ "#1"#2#3>{\_checkbrefD #2#3>}
1631 \_def\checkbrefD #1>{%
1632   \_isinlist{.#1}{<}\_iftrue\_checkbrefS#1>\_else\_checkbrefN#1>\_fi
1633 }
1634 \_def\checkbrefS #1 #2>{\_checkbrefN#2>}
1635 \_def\checkbrefN #1>{%
1636   \_def\tpmb{#1}
1637   \_ifx\tpmb\_empty \_errmessage{missing link data}\_nobref\_else
1638   \_replstring\tpmb{:}{\_replstring\tpmb{-}{\_replstring\tpmb{\_}{\_}%
1639   \_replstring\tpmb{a}{\_replstring\tpmb{b}{\_replstring\tpmb{c}{\_}%
1640   \_setbox0=\_hbox{\_tmpnum=0\tpmb\_relax}%
1641   \_ifdim\_wd0>0pt \_errmessage{nonnumeric link data}\_nobref\_fi
1642   \_fi
1643 }
1644 \_def\_nobref{\_def\_bref##1>{\_Red\_string<##1>}}
1645 \_def\currbook{}
1646 \_def\prelinkB{BK}
1647 \_def\prelinkC{BK}
1648 \_def\prelinkV{0}
1649 \_def\nochapbooks{BK}
1650 \_let\<=<
1651
1652 \_def\x/#1/{\_def\tpmb{#1}%
1653   \_isinlist\tpmb{x}\_iftrue \_badx
1654   \_else \_isinlist\tpmb<\_iftrue \_badx
1655   \_else \_isinlist\tpm\enditems\_iftrue \_badx \_else \_x/#1/\_fi\_fi\_fi
1656 }
1657 \_def\_badx{\_errmessage{unclosed \_string\x/.../}}
1658
1659 \_def\Article[#1]{}
1660 \_def\Cite #1 {\_par\_noindent{\_bf Cite: }}
1661 \_def\insertCite #1#2{}
1662
1663 \_def\putArticle #1 #2[#3]#4(#5){}
1664 \_def\putCite #1:#2 {\_par\_noindent{\_bf Cite: }}
1665 \_def\putBot #1 #2[#3]#4(#5){\_vbox}
1666
1667 \_def\c[#1/#2]#3{#3}
1668
1669 \_long\_ea\_def\_csname Note\_endcsname #1 #2#3%
1670
1671 {\_par \_let\nextww\_undefined \_noindent{\_bf Note #1:} #3\_par}
```

```

1672 }}
1673 \nspublic \checksyntax ;

```

## 16 TODO macros

The temporary macros are here. I plan to rewrite them.

op-bible.opm

```

1683 \_def\chaptit#1{\_line{\_hss\chapfont\Red#1\_hss}
1684 \_nobreak
1685 }
1686 \_def\schaptit#1{\_bigskip\chaptit{#1}\_medskip}
1687
1688 \nspublic \chaptit ;
1689
1690 \_sdef\_mt:intro:en}{Introduction} \_sdef\_mt:outline:en}{Outline}
1691 \_sdef\_mt:intro:cs}{Úvod} \_sdef\_mt:outline:cs}{Osnova}
1692
1693 \_def\dopsat{{\Red !!! DOPSAT !!! }}
1694
1695 \_def\Bibleinput#1 {\_bgroup
1696 \_catcode`##=13 \_bgroup\_lccode`~=`## \_lowercase{\_egroup\_let~}=\_processline
1697 \_input{#1}%
1698 \_egroup
1699 }
1700 \_let\FormattedBook=\_ignoreit % for backward compatibility
1701 \_let\CommentedBook=\_ignoreit % for backward compatibility

```

Active character < used for references.

op-bible.opm

```

1707 \_outer\_def\Note{\_Note}
1708
1709 \_def\_afterload{\_adeft<{\_bref}}
1710 \_afterload
1711
1712 \_endnamespace

```

## 17 Index

\.AddNote 5	\.currversenum 7–8	\.nextwwA 5
\alist! 4, 7	\.currversetext 7	\nochapbooks 3
\amark 2	\.doNote 4, 6	\Note 4–6
\.begblock 19	\encenter 7	\.NoteB 5
\begcenter 7	\.endblock 19	\.notefail 6
\bex! 2–3	\endcenter 7	\.notelog 6
\.bibleinput 2	\f! 3	\.notenum 5
\bibname 3	\fmtfile 2	\notepre! 5
\bmark 2–3	\fmtins 6	\noteref! 5
\BookException 2–3	\fmtpre 6–8	\notesfile 2
\BookPost 2–3	\.fmtprebuff 6	\notetext! 5
\BookPre 2–3	\ftmadd 6	\pbook! 2
\BookTile 3	\.fullvref 5	\.printbeforefirst 8
\bpo! 2–3	\.fullvrefm 5	\.printchap 7–8
\bpr! 2–3	\.gentovref 4	\printedbooks 2
\.brefBookChapter 3	\introfile 2, 19	\.printintro 19
\.btit 2	\.iscolonin 6	\.printverse 7–8
\btit! 3	\.isdivisin 6	\.printwarn 1
\.buff 4, 6–8	\.myaddto 1	\processbooks 2–3
\centeringmode 7	\.newaction 4–5, 7	\.processline 7
\.checknochapbooks 3	\.newbook 2–3	\.processverse 7
\.currbook 2, 4, 7	\nextww 4	\pword! 5
\.currchapnum 7	\.nextww 5	\renum 4–5
\.currverse 7	\nextwwA 4	\renumlabel 5–6



```
\.renumvref 4  
\.replpre 4  
\.sedef 1
```

```
\.setheadline 3  
\.transformword 4–5  
\.upcasefirst 6
```

```
\ww 4
```