

The **zref-vario** package^{*}

Code documentation

Gustavo Barros[†]

2022-01-31

EXPERIMENTAL

Contents

1	Initial setup	2
2	Customization	2
3	User interface	9
4	Languages	12
	Index	14

^{*}This file describes v0.1.0-alpha, released 2022-01-31.

[†]<https://github.com/gusbrs/zref-vario>

1 Initial setup

Start the DocStrip guards.

```
1 <*package>
   Identify the internal prefix (LATEX3 DocStrip convention).
2 <@@=zrefvario>
   Identify the package.
3 \ProvidesExplPackage {zref-vario} {2022-01-31} {0.1.0-alpha}
4   {Extended LATEX page cross-references with varioref and zref-clever}
   Required packages.
5 \RequirePackage { varioref }
6 \RequirePackage { zref-clever } [ 2022-01-28 ] % v0.2.0-alpha
```

2 Customization

Provide some functions to define options names in a standard form.

```
\_zrefvario_ref_opt_default_cs:nn      \_zrefvario_ref_opt_default_cs:nn {<option>}} {<signature>}}
7 \cs_new:Npn \_zrefvario_ref_opt_default_cs:nn #1#2
8   { \_zrefvario_ref_opt_default_ #1 : #2 }
(End definition for \_zrefvario_ref_opt_default_cs:nn.)
```

```
\_zrefvario_base_lang:n               \_zrefvario_base_lang:n {<language>}}
9 \cs_new:Npn \_zrefvario_base_lang:n #1
10  {
11    \zrefclever_language_if_declared:nTF {#1}
12      { \tl_use:c { \zrefclever_language_varname:n {#1} } }
13      { zv_unknown_language }
14  }
15 \cs_generate_variant:Nn \_zrefvario_base_lang:n { e }
(End definition for \_zrefvario_base_lang:n.)
```

\l_zrefvario_ref_language_tl An internal version of \l_zrefclever_ref_language_tl, since we need to retrieve the current value of the variable in multiple places.

```
16 \tl_new:N \l_zrefvario_ref_language_tl
17 \tl_set:Nn \l_zrefvario_ref_language_tl { \l_zrefclever_ref_language_tl }
(End definition for \l_zrefvario_ref_language_tl.)
```

```
\_zrefvario_ref_opt_name_lang_tl:nn   \_zrefvario_ref_opt_name_lang_tl:nn {<lang>}} {<option>}}
18 \cs_new:Npn \_zrefvario_ref_opt_name_lang_tl:nn #1#2
19   { g\_zrefvario_ref_opt_lang_ \_zrefvario_base_lang:n {#1} _ #2 _tl }
20 \cs_generate_variant:Nn \_zrefvario_ref_opt_name_lang_tl:nn { en }
(End definition for \_zrefvario_ref_opt_name_lang_tl:nn.)
```

```

\__zrefvario_ref_opt_name_lang_cs:nnn {\lang} {\option} {\signature}}
21 \cs_new:Npn \__zrefvario_ref_opt_name_lang_cs:nnn #1#2#3
22   { \__zrefvario_ref_opt_lang_ \__zrefvario_base_lang:n {#1} _ #2 : #3 }
23 \cs_generate_variant:Nn \__zrefvario_ref_opt_name_lang_cs:nnn { enn }

(End definition for \__zrefvario_ref_opt_name_lang_cs:nnn.)
Retrieving options values.

\__zrefvario_ref_opt_tl_get:nnN {\lang} {\option} {\tl var to set}}
\__zrefvario_ref_opt_cs_get:nnnN {\lang} {\option}
{\cs signature} {\cs to set}}
24 \cs_new_protected:Npn \__zrefvario_ref_opt_tl_get:nnN #1#2#3
25   {
26     \tl_if_exist:cTF
27       { \__zrefvario_ref_opt_name_lang_tl:nn {#1} {#2} }
28       {
29         \tl_set_eq:Nc #3
30         { \__zrefvario_ref_opt_name_lang_tl:nn {#1} {#2} }
31       }
32       {
33         \tl_set_eq:Nc #3
34         { \__zrefvario_ref_opt_name_lang_tl:nn { zv_unknown_language } {#2} }
35       }
36   }
37 \cs_generate_variant:Nn \__zrefvario_ref_opt_tl_get:nnN { xnN }
38 \cs_new_protected:Npn \__zrefvario_ref_opt_cs_get:nnnN #1#2#3#4
39   {
40     \cs_if_exist:cTF
41       { \__zrefvario_ref_opt_name_lang_cs:nnn {#1} {#2} {#3} }
42       {
43         \cs_set_eq:Nc #4
44         { \__zrefvario_ref_opt_name_lang_cs:nnn {#1} {#2} {#3} }
45       }
46       {
47         \cs_if_exist:cTF
48           { \__zrefvario_ref_opt_default_cs:nn {#2} {#3} }
49           {
50             \cs_set_eq:Nc #4
51             { \__zrefvario_ref_opt_default_cs:nn {#2} {#3} }
52           }
53           {
54             \cs_set_eq:Nc #4
55             {
56               \__zrefvario_ref_opt_name_lang_cs:nnn
57               { zv_unknown_language } {#2} {#3}
58             }
59           }
60       }
61   }
62 \cs_generate_variant:Nn \__zrefvario_ref_opt_cs_get:nnnN { xnnN }

(End definition for \__zrefvario_ref_opt_tl_get:nnN and \__zrefvario_ref_opt_cs_get:nnnN.)

```

Handle the options set of the user commands, and add support for varioref optional arguments. All commands can receive the same set of options \zcref can,

and these are processed the same way and “consumed” by any `\zceref` or `\zcpageref` calls. Besides those, `vother` is made available for `\zvpageref`, corresponding to its second optional argument, and `vcurrent` which is available to all commands except `\fullref`, corresponding to their first optional arguments. The distinction between `\l__zrefvario_reftextcurrent_tl` and `\l__zrefvario_vref_reftextcurrent_tl` is in their default/initial values. The default values for the optional arguments of `\vpageref`, `\vrefrange`, and `\vpagerefrange` are set to `\reftextcurrent`, while the default value of the optional argument of `\vref` is empty.

```

63 \tl_new:N \l__zrefvario_reftextcurrent_tl
64 \tl_set:Nn \l__zrefvario_reftextcurrent_tl { \reftextcurrent }
65 \tl_new:N \l__zrefvario_vref_reftextcurrent_tl
66 \tl_new:N \l__zrefvario_reftextother_tl
67 \keys_define:nn { zref-vario/current }
68 {
69     vcurrent .code:n =
70     {
71         \tl_set:Nn \l__zrefvario_reftextcurrent_tl {#1}
72         \tl_set:Nn \l__zrefvario_vref_reftextcurrent_tl {#1}
73     } ,
74     vcurrent .value_required:n = true ,
75 }
76 \keys_define:nn { zref-vario/other }
77 {
78     vother .tl_set:N = \l__zrefvario_reftextother_tl ,
79     vother .value_required:n = true ,
80 }
81 \keys_define:nn { }
82 {
83     zref-vario/ref .inherit:n =
84     { zref-clever/reference } ,
85     zref-vario/ref-oneoptarg .inherit:n =
86     { zref-clever/reference , zref-vario/current } ,
87     zref-vario/ref-twooptargs .inherit:n =
88     { zref-clever/reference , zref-vario/current , zref-vario/other } ,
89 }

```

`\zvLanguageSetup` User interface for setting “strings” (`reftext...` options) and “formats” (`...format` options) for each language.

```

\zvLanguageSetup{<language>}{<options>}

90 \tl_new:N \l__zrefvario_setup_language_tl
91 \seq_new:N \g__zrefvario_setup_languages_seq
92 \NewDocumentCommand \zvLanguageSetup { m m }
93 {
94     \group_begin:
95     \zrefclever_language_if_declared:nTF {#1}
96     {
97         \tl_set:Nn \l__zrefvario_setup_language_tl {#1}
98         \keys_set:nn { zref-vario/langsetup } {#2}
99         \seq_gput_right:Nx \g__zrefvario_setup_languages_seq
100         { \__zrefvario_base_lang:n {#1} }
101     }

```

```

102     { \msg_warning:nnn { zref-vario } { unknown-language-setup } {#1} }
103   \group_end:
104 }
105 \msg_new:nnn { zref-vario } { unknown-language-setup }
106 {
107   Language~'~{#1}'~is~unknown~\msg_line_context:..~Can't~set~it~up.~
108   The~language~must~be~declared~for~'zref-clever',~see~its~documentation.
109 }
110 \@onlypreamble \zvLanguageSetup

(End definition for \zvLanguageSetup.)

111 \keys_define:nn { zref-vario/langsetup }
112 {
113   retextfaceafter .code:n =
114   {
115     \tl_gset:cn
116     {
117       \__zrefvario_ref_opt_name_lang_tl:en
118       { \l__zrefvario_setup_language_tl }
119       { retextfaceafter }
120     } {#1}
121   } ,
122   retextfacebefore .code:n =
123   {
124     \tl_gset:cn
125     {
126       \__zrefvario_ref_opt_name_lang_tl:en
127       { \l__zrefvario_setup_language_tl }
128       { retextfacebefore }
129     } {#1}
130   } ,
131   retextafter .code:n =
132   {
133     \tl_gset:cn
134     {
135       \__zrefvario_ref_opt_name_lang_tl:en
136       { \l__zrefvario_setup_language_tl }
137       { retextafter }
138     } {#1}
139   } ,
140   retextbefore .code:n =
141   {
142     \tl_gset:cn
143     {
144       \__zrefvario_ref_opt_name_lang_tl:en
145       { \l__zrefvario_setup_language_tl }
146       { retextbefore }
147     } {#1}
148   } ,
149   retextcurrent .code:n =
150   {
151     \tl_gset:cn
152     {
153       \__zrefvario_ref_opt_name_lang_tl:en

```

```

154         { \l__zrefvario_setup_language_tl }
155         { reftextcurrent }
156     } {#1}
157 },
158 reftextfaraway .code:n =
159 {
160     \cs_gset_nopar:cpn
161     {
162         \__zrefvario_ref_opt_name_lang_cs:enn
163         { \l__zrefvario_setup_language_tl }
164         { reftextfaraway } { n }
165     }
166     ##1
167     {#1}
168 },
169 reftextpagerange .code:n =
170 {
171     \cs_gset_nopar:cpn
172     {
173         \__zrefvario_ref_opt_name_lang_cs:enn
174         { \l__zrefvario_setup_language_tl }
175         { reftextpagerange } { nn }
176     }
177     ##1##2
178     {#1}
179 },
180 reftextlabelrange .code:n =
181 {
182     \cs_gset_nopar:cpn
183     {
184         \__zrefvario_ref_opt_name_lang_cs:enn
185         { \l__zrefvario_setup_language_tl }
186         { reftextlabelrange } { nn }
187     }
188     ##1##2
189     {#1}
190 },
191 vrefformat .code:n =
192 {
193     \cs_gset_nopar:cpn
194     {
195         \__zrefvario_ref_opt_name_lang_cs:enn
196         { \l__zrefvario_setup_language_tl }
197         { vrefformat } { nn }
198     }
199     ##1##2
200     {#1}
201 },
202 vrefrangeformat .code:n =
203 {
204     \cs_gset_nopar:cpn
205     {
206         \__zrefvario_ref_opt_name_lang_cs:enn
207         { \l__zrefvario_setup_language_tl }

```

```

208         { vrefrangeformat } { nnn }
209     }
210     ##1##2##3
211     {#1}
212 } ,
213 fullreformat .code:n =
214 {
215     \cs_gset_nopar:cpn
216     {
217         \__zrefvario_ref_opt_name_lang_cs:enn
218         { \l__zrefvario_setup_language_tl }
219         { fullreformat } { n }
220     }
221     ##1
222     {#1}
223 } ,
224 }

```

Default reference formats for all languages. These will be used by `__zrefvario_ref_opt_cs_get:nnnN`, unless a language specific setting for the same format has been set. `\zreftextfaraway` is defined so it can be called with different arguments in `\zvpageref` (where it is alone) and in `\zfullref` (where it is the second of the pair).

```

225 \cs_new_nopar:cpn
226 { \__zrefvario_ref_opt_default_cs:nn { vrefformat } { nn } } #1#2
227 {
228     \zcref {#2} ~
229     \zvpageref [ S=false, typeset=both, vcurrent={#1} ] {#2}
230 }
231 \cs_new_nopar:cpn
232 { \__zrefvario_ref_opt_default_cs:nn { vrefrangeformat } { nnn } } #1#2#3
233 {
234     \reftextlabelrange {#2} {#3} ~
235     \zvpagerefrange [ S=false, typeset=both, vcurrent={#1} ] {#2} {#3}
236 }
237 \cs_new_nopar:cpn
238 { \__zrefvario_ref_opt_default_cs:nn { fullreformat } { n } } #1
239 {
240     \zcref {#1} ~
241     \zreftextfaraway [ S=false, typeset=both ] {#1}
242 }
243 \NewDocumentCommand \zreftextfaraway { s O { } m }
244 { \zref@wrapper@babel \__zrefvario_zreftextfaraway:nnn {#3} {#1} {#2} }
245 \cs_new_protected:Npn \__zrefvario_zreftextfaraway:nnn #1#2#3
246 {
247     \group_begin:
248     \keys_set:nn { zref-vario/ref } {#3}
249     \bool_if:nT {#2}
250     { \keys_set:nn { zref-vario/ref } { nohyperref } }
251     \__zrefvario_varioref_setup:
252     \reftextfaraway {#1}
253     \group_end:
254 }

```

Warnings and values for missing strings / unknown languages.

```

255 \group_begin:
256 \tl_set:Nn \l__zrefvario_setup_language_tl { zv_unknown_language }
257 \keys_set:nn { zref-vario/langsetup }
258 {
259   retextfaceafter =
260   {
261     \msg_warning:nxxx { zref-vario } { missing-string }
262     { retextfaceafter } { \l__zrefvario_ref_language_tl }
263     \zref@default
264   } ,
265   retextfacebefore =
266   {
267     \msg_warning:nxxx { zref-vario } { missing-string }
268     { retextfacebefore } { \l__zrefvario_ref_language_tl }
269     \zref@default
270   } ,
271   retextafter =
272   {
273     \msg_warning:nxxx { zref-vario } { missing-string }
274     { retextafter } { \l__zrefvario_ref_language_tl }
275     \zref@default
276   } ,
277   retextbefore =
278   {
279     \msg_warning:nxxx { zref-vario } { missing-string }
280     { retextbefore } { \l__zrefvario_ref_language_tl }
281     \zref@default
282   } ,
283   retextcurrent =
284   {
285     \msg_warning:nxxx { zref-vario } { missing-string }
286     { retextcurrent } { \l__zrefvario_ref_language_tl }
287     \zref@default
288   } ,
289   retextfaraway =
290   {
291     \msg_warning:nxxx { zref-vario } { missing-string }
292     { retextfaraway } { \l__zrefvario_ref_language_tl }
293     \zref@default
294   } ,
295   retextpagerange =
296   {
297     \msg_warning:nxxx { zref-vario } { missing-string }
298     { retextpagerange } { \l__zrefvario_ref_language_tl }
299     \zref@default
300   } ,
301   retextlabelrange =
302   {
303     \msg_warning:nxxx { zref-vario } { missing-string }
304     { retextlabelrange } { \l__zrefvario_ref_language_tl }
305     \zref@default
306   } ,
307 }
308 \group_end:

```



```

309 \msg_new:nnn { zref-vario } { missing-string }
310 { Missing~'#1'~string-for~language~'#2'~\msg_line_context:. }

A general initial warning in case the language is altogether not set up.

311 \AddToHook { begindocument }
312 {
313   \seq_if_in:NxF \g__zrefvario_setup_languages_seq
314   { \__zrefvario_base_lang:e { \l__zrefvario_ref_language_tl } }
315   {
316     \msg_warning:nnx { zref-vario } { language-not-setup }
317     { \l__zrefvario_ref_language_tl }
318   }
319 }
320 \msg_new:nnn { zref-vario } { language-not-setup }
321 {
322   Language~'#1'~is-not-set-up-for~'zref-vario'.~
323   See~documentation~for~'\iow_char:N\zvLanguageSetup'.
324 }

```

3 User interface

`__zrefvario_vref_pagenum:Nn` These three small functions are responsible for “connecting” `varioref` with `zref`, instead of with the standard referencing system. They are meant to locally replace their `varioref` counterparts: `\vref@pagenum`, `\vrefpagenum`, and `\vref@label`.

```

325 \cs_new_protected:Npn \__zrefvario_vref_pagenum:Nn #1#2
326 {
327   \exp_args:NNNo \exp_args:NNo \tl_set:Nn #1
328   { \zref@extractdefault {#2} { page } { ?? } }
329 }
330 \cs_new_protected:Npn \__zrefvario_vrefpagenum:Nn #1#2
331 {
332   \zref@refused {#2}
333   \__zrefvario_vref_pagenum:Nn #1 {#2}
334 }
335 \cs_new_protected:Npn \__zrefvario_vref_label:n #1
336 { \zref@labelbyprops {#1} { page } }

```

(End definition for `__zrefvario_vref_pagenum:Nn`, `__zrefvario_vrefpagenum:Nn`, and `__zrefvario_vref_label:n`.)

`__zrefvario_varioref_setup:` Sets things up for `varioref` to work with `zref/zref-clever`. It is pretty straightforward, but it is the core of the package. `\l__zrefvario_varioref_setup_bool` ensures this setup runs only once in case of nested calls to the package’s commands.

```

337 \bool_new:N \l__zrefvario_varioref_setup_bool
338 \cs_new_protected:Npn \__zrefvario_varioref_setup:
339 {
340   \bool_if:NF \l__zrefvario_varioref_setup_bool
341   {
342     \bool_set_true:N \l__zrefvario_varioref_setup_bool
343     \@vrefhandlespacefalse
344     \cs_set_eq:NN \vref@pagenum \__zrefvario_vref_pagenum:Nn
345     \cs_set_eq:NN \vrefpagenum \__zrefvario_vrefpagenum:Nn
346     \cs_set_eq:NN \vref@label \__zrefvario_vref_label:n

```

```

347 \_zrefvario_ref_opt_tl_get:xnN
348 { \l\_zrefvario_ref_language_tl }
349 { retextfaceafter }
350 \retextfaceafter
351 \_zrefvario_ref_opt_tl_get:xnN
352 { \l\_zrefvario_ref_language_tl }
353 { retextfacebefore }
354 \retextfacebefore
355 \_zrefvario_ref_opt_tl_get:xnN
356 { \l\_zrefvario_ref_language_tl }
357 { retextafter }
358 \retextafter
359 \_zrefvario_ref_opt_tl_get:xnN
360 { \l\_zrefvario_ref_language_tl }
361 { retextbefore }
362 \retextbefore
363 \_zrefvario_ref_opt_tl_get:xnN
364 { \l\_zrefvario_ref_language_tl }
365 { retextcurrent }
366 \retextcurrent
367 \_zrefvario_ref_opt_cs_get:xnnN
368 { \l\_zrefvario_ref_language_tl }
369 { retextfaraway } { n }
370 \retextfaraway
371 \_zrefvario_ref_opt_cs_get:xnnN
372 { \l\_zrefvario_ref_language_tl }
373 { retextpagerange } { nn }
374 \retextpagerange
375 \_zrefvario_ref_opt_cs_get:xnnN
376 { \l\_zrefvario_ref_language_tl }
377 { retextlabelrange } { nn }
378 \retextlabelrange
379 \_zrefvario_ref_opt_cs_get:xnnN
380 { \l\_zrefvario_ref_language_tl }
381 { vrefformat } { nn }
382 \vrefformat
383 \_zrefvario_ref_opt_cs_get:xnnN
384 { \l\_zrefvario_ref_language_tl }
385 { vrefrangeformat } { nnn }
386 \vrefrangeformat
387 \_zrefvario_ref_opt_cs_get:xnnN
388 { \l\_zrefvario_ref_language_tl }
389 { fullrefformat } { n }
390 \fullrefformat
391 }
392 }

```

(End definition for _zrefvario_varioref_setup:.)

User commands.

```

\zvref \zvref<*>[<options>]{<label>}
393 \NewDocumentCommand \zvref { s O { } m }
394 { \zref@wrapper@babel \_zrefvario_zvref:nnn {#3} {#1} {#2} }
395 \cs_new_protected:Npn \_zrefvario_zvref:nnn #1#2#3

```

```

396 {
397   \group_begin:
398   \keys_set:nn { zref-vario/ref-optional } {#3}
399   \bool_if:nT {#2}
400     { \keys_set:nn { zref-vario/ref } { nohyperref } }
401   \__zrefvario_varioref_setup:
402   \__zrefvario_vref:Vn \l__zrefvario_vref_reftextcurrent_tl {#1}
403   \group_end:
404 }

```

“Ideally”, we’d be using `\vref` here, but `hyperref` redefines `\vref@star` and `\vr@f`, hard-coding what was supposed to be configurable... , so we use `\vrefformat` instead, which is the original definition of `\vref@star`, and `\vr@f` also boils down to `\vref@star` when `\@vrefhandlespace` is false, which is our case.

```

405 \cs_new_protected:Npn \__zrefvario_vref:nn #1#2
406   { \vrefformat {#1} {#2} }
407 \cs_generate_variant:Nn \__zrefvario_vref:nn { Vn }

```

(End definition for `\zvref`.)

```

\zvpageref      \zvpageref{<*>[<options>]}{<label>}}
408 \NewDocumentCommand \zvpageref { s O { } m }
409   { \zref@wrapper@babel \__zrefvario_zvpageref:nnn {#3} {#1} {#2} }
410 \cs_new_protected:Npn \__zrefvario_zvpageref:nnn #1#2#3
411   {
412     \group_begin:
413     \keys_set:nn { zref-vario/ref-twooptargs } {#3}
414     \bool_if:nT {#2}
415       { \keys_set:nn { zref-vario/ref } { nohyperref } }
416     \__zrefvario_varioref_setup:
417     \__zrefvario_vpageref:VVn \l__zrefvario_reftextcurrent_tl
418       \l__zrefvario_reftextother_tl {#1}
419     \group_end:
420   }
421 \cs_new_protected:Npn \__zrefvario_vpageref:nnn #1#2#3
422   { \pageref [{#1}] [{#2}] {#3} }
423 \cs_generate_variant:Nn \__zrefvario_vpageref:nnn { VVn }

```

(End definition for `\zvpageref`.)

```

\zvrefrange     \zvrefrange{<*>[<options>]}{<label>}{<label>}}
424 \NewDocumentCommand \zvrefrange { s O { } m m }
425   {
426     \zref@wrapper@babel
427     { \zref@wrapper@babel \__zrefvario_zvrefrange:nnnn {#3} }
428     {#4} {#1} {#2}
429   }
430 \cs_new_protected:Npn \__zrefvario_zvrefrange:nnnn #1#2#3#4
431   {
432     \group_begin:
433     \keys_set:nn { zref-vario/ref-optional } {#4}
434     \bool_if:nT {#3}
435       { \keys_set:nn { zref-vario/ref } { nohyperref } }
436     \__zrefvario_varioref_setup:

```

```

437 \__zrefvario_vrefrange:Vnn \l__zrefvario_reftextcurrent_tl {#1} {#2}
438 \group_end:
439 }
440 \cs_new_protected:Npn \__zrefvario_vrefrange:nnn #1#2#3
441 { \vrefrange [{#1}] {#2} {#3} }
442 \cs_generate_variant:Nn \__zrefvario_vrefrange:nnn { Vnn }

(End definition for \zvrefrange.)

```

```

\zvpagerefrange \zvpagerefrange{*}[<options>]{<label>}{<label>}
443 \NewDocumentCommand \zvpagerefrange { s O { } m m }
444 {
445 \zref@wrapper@babel
446 { \zref@wrapper@babel \__zrefvario_zvpagerefrange:nnnn {#3} }
447 {#4} {#1} {#2}
448 }
449 \cs_new_protected:Npn \__zrefvario_zvpagerefrange:nnnn #1#2#3#4
450 {
451 \group_begin:
452 \keys_set:nn { zref-vario/ref-oneoptarg } {#4}
453 \bool_if:nT {#3}
454 { \keys_set:nn { zref-vario/ref } { nohyperref } }
455 \__zrefvario_varioref_setup:
456 \__zrefvario_vpagerefrange:Vnn \l__zrefvario_reftextcurrent_tl {#1} {#2}
457 \group_end:
458 }
459 \cs_new_protected:Npn \__zrefvario_vpagerefrange:nnn #1#2#3
460 { \vpagerefrange [{#1}] {#2} {#3} }
461 \cs_generate_variant:Nn \__zrefvario_vpagerefrange:nnn { Vnn }

(End definition for \zvpagerefrange.)

```

```

\zfullref \zfullref{*}[<options>]{<label>}
462 \NewDocumentCommand \zfullref { s O { } m }
463 { \zref@wrapper@babel \__zrefvario_zfullref:nnn {#3} {#1} {#2} }
464 \cs_new_protected:Npn \__zrefvario_zfullref:nnn #1#2#3
465 {
466 \group_begin:
467 \keys_set:nn { zref-vario/ref } {#3}
468 \bool_if:nT {#2}
469 { \keys_set:nn { zref-vario/ref } { nohyperref } }
470 \__zrefvario_varioref_setup:
471 \fullref {#1}
472 \group_end:
473 }

(End definition for \zfullref.)

```

4 Languages

```

474 \zvLanguageSetup { english }
475 {
476 reftextfaceafter = {on~the~\reftextvario{facing}{next}~page} ,

```

```

477     retextfacebefore = {on~the~\retextvario{facing}{preceding}~page} ,
478     retextafter = {on~the~\retextvario{following}{next}~page} ,
479     retextbefore = {on~the~\retextvario{preceding}{previous}~page} ,
480     retextcurrent = {on~\retextvario{this}{the~current}~page} ,
481     retextfaraway = {on~\zcpageref{#1}} ,
482     retextpagerange = {on~\zcpageref[range]{#1,#2}} ,
483     retextlabelrange = {\zcref[range]{#1,#2}} ,
484 }

485 \zvLanguageSetup { german }
486 {
487     retextfaceafter = {auf~der~nächsten~Seite} ,
488     retextfacebefore = {auf~der~vorherigen~Seite} ,
489     retextafter = {auf~der~nächsten~Seite} ,
490     retextbefore = {auf~der~vorherigen~Seite} ,
491     retextcurrent = {auf~dieser~Seite} ,
492     retextfaraway = {auf~\zcpageref[d=D,g=f]{#1}} ,
493     retextpagerange = {auf~den~\zcpageref[range,d=D,g=f]{#1,#2}} ,
494     retextlabelrange = {\zcref[range]{#1,#2}} ,
495 }

496 \zvLanguageSetup { french }
497 {
498     retextfaceafter = {page~\retextvario{ci-contre}{suivante}} ,
499     retextfacebefore = {page~\retextvario{ci-contre}{précédente}} ,
500     retextafter = {page~suivante} ,
501     retextbefore = {page~précédente} ,
502     retextcurrent = {de~la~présente~page} ,
503     retextfaraway = {\zcpageref[g=f]{#1}} ,
504     retextpagerange = {\zcpageref[range,g=f]{#1,#2}} ,
505     retextlabelrange = {\zcref[range]{#1,#2}} ,
506 }

507 \zvLanguageSetup { portuguese }
508 {
509     retextfaceafter = {na~\retextvario{página-oposta}{próxima-página}} ,
510     retextfacebefore = {na~página~\retextvario{oposta}{anterior}} ,
511     retextafter = {na~\retextvario{página-seguinte}{próxima-página}} ,
512     retextbefore = {na~página~\retextvario{anterior}{precedente}} ,
513     retextcurrent = {nesta~página} ,
514     retextfaraway = {na~\zcpageref[g=f]{#1}} ,
515     retextpagerange = {nas~\zcpageref[range,g=f]{#1,#2}} ,
516     retextlabelrange = {\zcref[range]{#1,#2}} ,
517 }

518 \zvLanguageSetup { spanish }
519 {
520     retextfaceafter = {en~la~página~siguiente} ,
521     retextfacebefore = {en~la~página~anterior} ,
522     retextafter = {en~la~página~siguiente} ,
523     retextbefore = {en~la~página~anterior} ,
524     retextcurrent = {en~esta~página} ,
525     retextfaraway = {en~la~\zcpageref[g=f]{#1}} ,
526     retextpagerange = {en~las~\zcpageref[range,g=f]{#1,#2}} ,
527     retextlabelrange = {\zcref[range]{#1,#2}} ,
528 }

```

```

529 \zvLanguageSetup { dutch }
530 {
531     retextfaceafter = {op-de~\reftextvario{rechter~pagina}{pagina~hiernaast}} ,
532     retextfacebefore = {op-de~\reftextvario{linker~pagina}{pagina~hiernaast}} ,
533     retextafter = {op-de~\reftextvario{volgende~pagina}{pagina~hierna}} ,
534     retextbefore = {op-de~\reftextvario{vorige~pagina}{pagina~hiervoor}} ,
535     retextcurrent = {op-deze~pagina} ,
536     retextfaraway = {op~\zcpageref[g=f]{#1}} ,
537     retextpagerange = {op~\zcpageref[range,g=f]{#1,#2}} ,
538     retextlabelrange = {\zcref[range]{#1,#2}} ,
539 }
540 </package>

```

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

A		G	
\AddToHook	311	group commands:	
		\group_begin:	94, 247, 255, 397, 412, 432, 451, 466
		\group_end:	103, 253, 308, 403, 419, 438, 457, 472
B		I	
bool commands:		iow commands:	
\bool_if:NTF	340	\iow_char:N	323
\bool_if:nTF	249, 399, 414, 434, 453, 468		
\bool_new:N	337		
\bool_set_true:N	342		
C		K	
cs commands:		keys commands:	
\cs_generate_variant:Nn	15, 20, 23, 37, 62, 407, 423, 442, 461	\keys_define:nn	67, 76, 81, 111
\cs_gset_nopar:Npn	160, 171, 182, 193, 204, 215	\keys_set:nn	98, 248, 250, 257, 398, 400, 413, 415, 433, 435, 452, 454, 467, 469
\cs_if_exist:NTF	40, 47		
\cs_new:Npn	7, 9, 18, 21	M	
\cs_new_nopar:Npn	225, 231, 237	msg commands:	
\cs_new_protected:Npn	24, 38, 245, 325, 330, 335, 338, 395, 405, 410, 421, 430, 440, 449, 459, 464	\msg_line_context:	107, 310
\cs_set_eq:NN	43, 50, 54, 344, 345, 346	\msg_new:nnn	105, 309, 320
		\msg_warning:nnn	102, 316
		\msg_warning:nnnn	261, 267, 273, 279, 285, 291, 297, 303
E		N	
exp commands:		\NewDocumentCommand	92, 243, 393, 408, 424, 443, 462
\exp_args:NNNo	327		
\exp_args:NNo	327	P	
F		\ProvidesExplPackage	3
\fullref	4, 471	R	
\fullreformat	390	\retextafter	358

<code>\reftextbefore</code>	362	<code>\zcref</code>	3, 4, 228, 240, 483, 494, 505, 516, 527, 538
<code>\reftextcurrent</code>	4, 64, 366	<code>\zfullref</code>	7, 12, 462
<code>\reftextfaceafter</code>	350	zrefclever commands:	
<code>\reftextfacebefore</code>	354	<code>\zrefclever_language_if_declared:nTF</code>	11, 95
<code>\reftextfaraway</code>	252, 370	<code>\zrefclever_language_varname:n</code> ..	12
<code>\reftextlabelrange</code>	234, 378	<code>\l_zrefclever_ref_language_tl</code> ..	2, 17
<code>\reftextpagerange</code>	374	<code>\zrefcleverfaraway</code>	7, 241, 243
<code>\reftextvario</code>	476, 477, 478, 479, 480, 498, 499, 509, 510, 511, 512, 531, 532, 533, 534	zrefvario internal commands:	
<code>\RequirePackage</code>	5, 6	<code>__zrefvario_base_lang:n</code>	2, 9, 19, 22, 100, 314
S			
seq commands:			
<code>\seq_gput_right:Nn</code>	99	<code>\l__zrefvario_ref_language_tl</code> ...	16, 262, 268, 274, 280, 286, 292, 298, 304, 314, 317, 348, 352, 356, 360, 364, 368, 372, 376, 380, 384, 388
<code>\seq_if_in:NnTF</code>	313	<code>__zrefvario_ref_opt_cs_get:nnnN</code> ..	3, 7, 24, 367, 371, 375, 379, 383, 387
<code>\seq_new:N</code>	91	<code>__zrefvario_ref_opt_default_- cs:nn</code>	2, 7, 48, 51, 226, 232, 238
T			
TeX and L ^A T _E X 2 _ε commands:			
<code>\@onlypreamble</code>	110	<code>__zrefvario_ref_opt_name_lang_- cs:nnn</code>	3, 21, 41, 44, 56, 162, 173, 184, 195, 206, 217
<code>\@vrefhandlespace</code>	11	<code>__zrefvario_ref_opt_name_lang_- tl:nn</code>	2, 18, 27, 30, 34, 117, 126, 135, 144, 153
<code>\@vrefhandlespacefalse</code>	343	<code>__zrefvario_ref_opt_tl_get:nnN</code> ..	3, 24, 347, 351, 355, 359, 363
<code>\vr@f</code>	11	<code>\l__zrefvario_ref_textcurrent_tl</code> ..	4, 63, 64, 71, 417, 437, 456
<code>\vref@label</code>	9, 346	<code>\l__zrefvario_ref_textother_tl</code> ...	66, 78, 418
<code>\vref@pagenum</code>	9, 344	<code>\l__zrefvario_setup_language_tl</code> ..	90, 97, 118, 127, 136, 145, 154, 163, 174, 185, 196, 207, 218, 256
<code>\vref@star</code>	11	<code>\g__zrefvario_setup_languages_- seq</code>	91, 99, 313
<code>\zref@default</code>	263, 269, 275, 281, 287, 293, 299, 305	<code>__zrefvario_varioref_setup:</code> ...	251, 337, 401, 416, 436, 455, 470
<code>\zref@extractdefault</code>	328	<code>\l__zrefvario_varioref_setup_- bool</code>	9, 337, 340, 342
<code>\zref@labelbyprops</code>	336	<code>__zrefvario_vpageref:nnn</code>	417, 421, 423
<code>\zref@refused</code>	332	<code>__zrefvario_vpagerefrange:nnn</code> ..	456, 459, 461
<code>\zref@wrapper@babel</code>	244, 394, 409, 426, 427, 445, 446, 463	<code>__zrefvario_vref:nn</code> ..	402, 405, 407
tl commands:			
<code>\tl_gset:Nn</code> ...	115, 124, 133, 142, 151	<code>__zrefvario_vref_label:n</code> ..	325, 346
<code>\tl_if_exist:NnTF</code>	26	<code>__zrefvario_vref_pagenum:Nn</code> ..	325, 344
<code>\tl_new:N</code>	16, 63, 65, 66, 90	<code>\l__zrefvario_vref_ref_textcurrent_- tl</code>	4, 65, 72, 402
<code>\tl_set:Nn</code> ..	17, 64, 71, 72, 97, 256, 327	<code>__zrefvario_vref_pagenum:Nn</code> ..	325, 345
<code>\tl_set_eq:NN</code>	29, 33	<code>__zrefvario_vrefrange:nnn</code>	437, 440, 442
<code>\tl_use:N</code>	12	<code>__zrefvario_zfullref:nnn</code> ..	463, 464
V			
<code>\vpageref</code>	4, 422		
<code>\vpagerefrange</code>	4, 460		
<code>\vref</code>	4, 11		
<code>\vrefformat</code>	11, 382, 406		
<code>\vrefpagenum</code>	9, 345		
<code>\vrefrange</code>	4, 441		
<code>\vrefrangeformat</code>	386		
Z			
<code>\zcpageref</code>	4, 481, 482, 492, 493, 503, 504, 514, 515, 525, 526, 536, 537		

__zrefvario_zreftextfaraway:nnn	\zvLanguageSetup
..... 244, 245	... 4, 90, 474, 485, 496, 507, 518, 529
__zrefvario_zvpageref:nnn . 409, 410	\zvpageref 4, 7, 11, 229, 408
__zrefvario_zvpagerefrange:nnnn	\zvpagerefrange 12, 235, 443
..... 446, 449	
__zrefvario_zvref:nnn 394, 395	\zvref 10, 393
__zrefvario_zvrefrange:nnnn 427, 430	\zvrefrange 11, 424