

LuaQuotes: A package for smart quotation marks

Elijah Z Granet*

15 January 2023

Version 1.3.0

Contents

1 Overview	3
1.1 Purpose	3
2 Smart quotes	3
2.1 Options	3
2.1.1 English Features	3
2.1.2 French option	4
2.1.3 The German option	4
2.1.4 The German (degm) option	4
2.2 Activation and De-activation	4
2.3 Monospace	5
3 Auxiliary Punctuation	5
3.1 Standalone quotes	5
3.2 Additional symbols	8
4 Future Development and Localisation	8
5 Licence	9
6 Implementation	10
7 Version History	20
7.1 1.3.0	20

*e-mail: ezg21@cantab.ac.uk

1 Overview

1.1 Purpose

This package provides a function to automatically have ‘smart quotes’ in Lua^LA^TE^X. By ‘smart quotes’, I refer to the automatic insertion of curved or ‘typographer’s’ quotation marks when the user types straight quotation marks. The below figure illustrates the distinction in English typography:

User input	Smart Quotes Off	Smart Quotes On
"Howdy!"	”Howdy!”	“Howdy!”
'Don't!'	’Don’t!’	‘Don’t!’

2 Smart quotes

2.1 Options

The default option, for English quotation marks, is called by:

```
\usepackage{luaquotes}
```

The French and German options, for those languages’ punctuation, are called by the following respective commands.

```
\usepackage[fr]{luaquotes} %French  
\usepackage[de]{luaquotes} % German  
\usepackage[degm]{luaquotes} % German with French-style quotes
```

2.1.1 English Features

The English features are designed to smartly recognise English punctuation:

Feature	User input	Output
Double Quotes	"Salut!"	“Hello”
Single Quotes	'Hello!'	‘Hello’
Contractions	Don't	Don’t

2.1.2 French option

The French option produces the following output, including the extra space around punctuation prescribed by French typography:

User input	Output
"Salut!"	« Salut! »
'Salut!'	< Salut! >

2.1.3 The German option

The German option produces the following output:

User input	Output
"Hallo!"	„Hallo!“
'Hallo!'	,Hallo!‘

2.1.4 The German (degm) option

The German (degm) option produces the following output:

User input	Output
"Hallo!"	»Hallo!«
'Hallo!'	>Hallo!<

2.2 Activation and De-activation

The package automatically activates the smart quotes function at the beginning of the document. To deactivate the smart quotes function within a document, the following commands are used:

```
\dumbquotes %English  
\frdumbquotes %French  
\dedumbquotes %German  
\degmdumbquotes %German with French-style quotes
```

The following commands re-activate the smart quotes function:

```
\smartquotes %English
\frsmartquotes %French
\desmartquotes % German
\degmsmartquotes % German with French-style quotes
```

A limitation on the (de-)activation of the package is that the Lua filters will not deactivate within the same paragraph, so the function can only be changed across paragraphs.

2.3 Monospace

As a general rule, smart quotes are rather undesirable in monospace text, and therefore, within the `\texttt` environment the package does not apply smart quotes. Thus, the same input produces in roman face “Hello World” but in monospace "Hello World".

As the example above shows, the default behaviour of this package forces straight quotes in monospace, and disables \TeX quote ligatures (but not other \TeX ligatures) to do so, on the assumption that any form of curved quotes are undesirable.

For extended periods of monospaced text called by `\ttfamily`, the activation and de-activation methods above should be used. I considered altering the `\ttfamily` command to always call on `\dumbquotes`, but I thought it best to avoid messing with the command. This may change in future development.

If a user desires to disable the smart quotes for other faces, this is easily done globally by adding the following line to the font’s configuration in `fontspec`:

```
RawFeature={+qtbye}
```

This will, however, result in the font using straight quotes rather than the standard \TeX quote ligatures, making it a distinct option than the activation and de-activation options *supra*.

3 Auxiliary Punctuation

The smart quotes feature covers the ‘standard’ usage of quotes, but there are many instances where quotation mark or quote-mark like features are needed outside the automatic formatting. The package provides several commands for this.

3.1 Standalone quotes

The marks in this section are the set of quotation marks used generally in writing and require little explanation. However, it should be noted that the commands `\sqoneright` and `\apost` (which produce identical output) are very

useful for aphetic words by which the first syllable is clipped, as in the sequence 'bout the smart quotes function will incorrectly produce an opening quote instead of the correct closing quote to indicate the elision. Thus, the incorrect result of 'bout is produced. The solution is to use the code \apost bout which produces 'bout.

Name	UTF-8	Command	Produces
Single low quote	U+201A	\desingle	, □
Double low quote	U+201E	\dedouble	,, □
Single straight quote	U+0027	\dqone	' □
Double straight quote	U+0022	\dqtwo	" □
Left single quote	U+2018	\sqoneleft	‘ □
Right single quote	U+2019	\sqoneright	□’
Apostrophe	U+2019	\apost	□’
Left double quote	U+201C	\sqtwoleft	“ □

Right double quote	U+201D	\sqtworight	”
Left guillemet [w/ space]	U+00AB	\glmtl	«
Right guillemet [w/ space]	U+00BB	\glmtr	»
degm left guillemet	U+00BB	\degmtl	»
degm right guillemet	U+00AB	\degmtr	«
Single left guillemet [w/ space]	U+2039	\sglmtl	<
Single right guillemet [w/ space]	U+203A	\sglmtr	>
degm single left guillemet	U+203A	\desgmtl	>
degm single right guillemet	U+2039	\desgmtr	<

3.2 Additional symbols

These are quote like symbols which are useful for precise punctuation, since standard smart quotation marks do not work well in their specialised use cases.

Name	UTF-8	Command	Produces
Backtick	U+0060	<code>\bcktick</code>	‘

For typesetting US/Imperial measurements like feet and inches, the correct symbol is a prime and double prime. Many modern typefaces have these symbols, and they are thus useful for typesetting feet and inch measurements (like 6’4”).

Name	UTF-8	Command	Produces
Single Prime	U+2032	<code>\lqprime</code>	′
Double Prime	U+2033	<code>\ldoubleprime</code>	″

For certain Polynesian languages, a letter called the ‘Okina is used; while this appears identical to a left single quotation mark in many fonts, it is encoded differently in Unicode because it is properly a letter, not a punctuation mark.

Name	UTF-8	Command	Produces
Okina	U+022B	<code>\okina</code>	‘

4 Future Development and Localisation

The package’s online repository is the best place to report bugs, feature requests, or other contributions, and is located at:
<https://github.com/ezgranet/luaquotes>.

One obvious point of future development is the addition of other language localisations; this is technologically easy in principle but difficult for me to accomplish without assistance because it requires detailed knowledge of typographic conventions for any given language standard, and therefore the aid of contributors and collaborators.

5 Licence

This project is licensed under the Latex Public Project Licence version 1.3c. This documentation is copyright of the author but licensed under CC-BY-SA 3.0.

6 Implementation

```
2
3 \def\luaquotesversionnumber{1.3.0}
4 \ProvidesPackage{luaquotes}
5   [2023/15/15\luaquotesversionnumber smart quotes with lua]
6   % !TeX program = lualatex
7   % !TeX encoding = utf8
8   % This work may be distributed and/or modified under the
9   % conditions of the LaTeX Project Public License, either version 1.3c
10  % of this license or (at your option) any later version.
11  % The latest version of this license is in
12  % http://www.latex-project.org/lppl.txt
13  % and version 1.3c or later is part of all distributions of LaTeX
14  % version 2005/12/01 or later.
15  %
16  % This work has the LPPL maintenance status `maintained'.
17  %
18  % The Current Maintainer of this work is Elijah Z Granet
19
20  %%%%%%%%%%%
21  %%%%%%%%%%%
22  % to show the package only works with Lua
23  %%%%%%%%%%%
24  %%%%%%%%%%%
25  \RequirePackage{iftex}
26  \ifPDFTeX {
27    \PackageError{luaquotes}
28      {You are using pdfTeX but this package only works
29      \MessageBreak with LuaTeX}{}
30  }
31  \else\ifXeTeX{ \PackageError{luaquotes}
32    {You are using XeTeX but this package only works
33    \MessageBreak with LuaTeX}{}
34  }\fi\fi
35  %%%%%%%%%%%
36  %%%%%%%%%%%
37  % Dependency
38  %%%%%%%%%%%
39  %%%%%%%%%%%
40  \RequirePackage{luacode}
41  %%%%%%%%%%%
42  %%%%%%%%%%%
43  % fontspec for the auxiliary
44  % quotes where tlig's need
45  % to be disabled
46  %%%%%%%%%%%
```

```

47 %%%%%%%%%%
48 \RequirePackage{fontspec}
49 %%%%%%%%%%
50
51 %%%%%%%%%%
52 %%%%%%%%%%
53 % removing the effects for monospace
54 %%%%%%%%%%
55 \begin{luacode}
56 local uchar = unicode.utf8.char
57   fonts.handlers.otf.addfeature{
58     name = "qtbye",
59     type = "substitution",
60     data =
61     {
62       " = 0x0022,
63       " = 0x0022,
64       ' = 0x0027,
65       « = 0x0022,
66       » = 0x0022,
67       < = 0x0027,
68       > = 0x0027
69     },
70   }
71 \end{luacode}
72 %%%%%%%%%%
73 %%%%%%%%%%
74 %%%%%%%%%%
75 %%%%%%%%%%
76 % Code here and throughout similar
77 %functions partly adapted from TeX.SE user
78 % Mico
79 %
80 ↪ https://tex.stackexchange.com/questions/499953/how-to-generate-correct-single-and-double-quotes-in-tex
81 %%%%%%%%%%
82
83 \luaexec{
84 function doublequotes ( s )
85   return ( s:gsub ( ““(.-)”” , ““\%1”” ) )
86   end}
87
88 %%%%%%%%%%
89 %%%%%%%%%%
90 % Assuming ‘ at the start of the line means an opening quotation mark not an
91 ↪ apostrophe
92 %%%%%%%%%%

```

```

93 \luaexec{function singlequotelinestart ( s )
94     return (s:gsub (“^’”,“‘” ) )
95     end}
96
97
98
99 % \luaexec{
100 % function abbrevsingle ( s )
101 %     return ( s:gsub ( “ ‘(..- ) ” , “ ’\%1 ” ) )
102 %     end}
103
104 \luaexec{function singlequotes ( s )
105     return ( s:gsub ( “ ‘”,“ ‘” ) )
106     end}
107
108
109 %%%%%%%%%%%
110 %%%%%%%%%%%
111 % parentheses
112 %%%%%%%%%%%
113 %%%%%%%%%%%
114 \newcommand{\sqoneleft}{\addfontfeature{RawFeature={-qtbye,-tlig}}‘}
115
116 \luaexec{function psinglequotes ( s )
117     return ( s:gsub ( “\%{\addfontfeature
118     ↪ {RawFeature={-qtbye,-tlig}}’”,[[(\sqoneleft]] ) )
119     end}
120
121 \newcommand\singlequotespon{\directlua{
122     luatexbase.add_to_callback (
123     “process_input_buffer” , psinglequotes , “psinglequotes” )
124     }}
125 \newcommand\singlequotespoff{\directlua{luatexbase.remove_from_callback (
126     “process_input_buffer” , “psinglequotes” )}}
127
128 %%%%%%%%%%%
129 %%%%%%%%%%%
130 % DE parentheses
131 %%%%%%%%%%%
132 %%%%%%%%%%%
133 %\newcommand{\sqoneleft}{\addfontfeature{RawFeature={-qtbye,-tlig}}‘}
134
135 \luaexec{function depsinglequotes ( s )
136     return ( s:gsub ( “\%{\addfontfeature
137     ↪ {RawFeature={-qtbye,-tlig}}’”,[[ (,)] ] ) )
138     end}

```

```

139 \newcommand\desinglequotespon{\directlua{
140 luatexbase.add_to_callback (
141     "process_input_buffer" , depsinglequotes , "depsinglequotes" )
142     }}
143 \newcommand\desinglequotespoff{\directlua{luatexbase.remove_from_callback (
144     "process_input_buffer" , "depsinglequotes" )}}
145
146 \luaexec{function depsinglequotesclose ( s )
147     return ( s:gsub ( "'\\%",[[\sqoneleft]] ) ) )
148     end}
149
150 \newcommand\desinglequotespcloseon{\directlua{
151 luatexbase.add_to_callback (
152     "process_input_buffer" , depsinglequotesclose , "depsinglequotesclose" )
153     }}
154 \newcommand\desinglequotespcloseoff{\directlua{luatexbase.remove_from_callback
155     ↵ (
156     "process_input_buffer" , "depsinglequotesclose" )}}
157
158
159 %%%%%%%%%%%
160 %%%%%%%%%%%
161 % FR Parentheses
162 %%%%%%%%%%%
163 %%%%%%%%%%%
164 \newcommand{\sglmltp}{\addfontfeature{RawFeature={-qtbye,-tlig}}\hskip
165     ↵ .16667em}}
166 \newcommand{\sglmltrp}{\addfontfeature{RawFeature={-qtbye,-tlig}}\hskip
167     ↵ .16667em}}
168
169 \luaexec{function frpsinglequotes ( s )
170     return ( s:gsub ( "%({\addfontfeature
171     ↵ {RawFeature={-qtbye,-tlig}}}'",[[\sglmltp]] ) )
172     end}
173
174 \luaexec{function frpsinglequotesclose ( s )
175     return ( s:gsub ( "'\\%",[[\sglmltrp]] ) ) )
176     end}
177
178 \newcommand\frsinglequotespon{\directlua{
179 luatexbase.add_to_callback (
180     "process_input_buffer" , frpsinglequotes , "frpsinglequotes" )
181     }}
182 \newcommand\frsinglequotespoff{\directlua{luatexbase.remove_from_callback (
183     "process_input_buffer" , "frpsinglequotes" )}}
184 \newcommand\frsinglequotespcloseon{\directlua{
185 luatexbase.add_to_callback (
186     "process_input_buffer" , frpsinglequotesclose , "frpsinglequotesclose" )
187     }}

```

```

183     }}
184 \newcommand\frsinglequotespcloseoff{\directlua{luatexbase.remove_from_callback
↪ (
185     "process_input_buffer" , "frsinglequotesclose" )}}
186
187
188 %%%%%%%%%%%
189 %%%%%%%%%%%
190 % activation and deactivation
191 %%%%%%%%%%%
192 %%%%%%%%%%%
193 \newcommand\doublequoteson{\directlua{luatexbase.add_to_callback (
194     "process_input_buffer" ,
195     doublequotes ,
196     "doublequotes" )}}
197 \newcommand\doublequotesoff{\directlua{luatexbase.remove_from_callback (
198     "process_input_buffer" ,
199     "doublequotes" )}}
200 % \newcommand\abbrevsingleon{\directlua{luatexbase.add_to_callback (
201 %     "process_input_buffer" ,
202 %     abbrevsingle ,
203 %     "abbrevsingle" )}}
204 % \newcommand\abbrevsingleoff{\directlua{luatexbase.remove_from_callback (
205 %     "process_input_buffer" ,
206 %     "abbrevsingle" )}}
207
208 \newcommand\singlequotelinestarton{\directlua{
209     luatexbase.add_to_callback (
210         "process_input_buffer" , singlequotelinestart , "singlequotelinestart" )
211     }}
212 \newcommand\singlequotelinestartoff{\directlua{
213     luatexbase.remove_from_callback (
214         "process_input_buffer" , "singlequotelinestart" )
215     }}
216 \newcommand\singlequoteson{\directlua{
217     luatexbase.add_to_callback (
218         "process_input_buffer" , singlequotes , "singlequotes" )
219     }}
220 \newcommand\singlequotesoff{\directlua{luatexbase.remove_from_callback (
221     "process_input_buffer" , "singlequotes" )}}
222 %%%%%%%%%%%
223 %%%%%%%%%%%
224 % global functions, useful for things like this
225 %%%%%%%%%%%
226 %%%%%%%%%%%
227 \newcommand{\smartquotes}{%
228 \doublequoteson%
229 \singlequotelinestarton%

```

```

230 % \abbrevsingleon%
231 \singlequoteson
232 \singlequotespon}
233 \newcommand{\dumbquotes}{
234 \doublequotesoff
235 % \abbrevsingleoff
236 \singlequotelinestartoff
237 \singlequotesoff
238 \singlequotespoff
239 }
240
241 \DeclareOption{en}{
242 \AtBeginDocument{\frsmartquotes\frdumbquotes%
243 \desmartquotes\dedumbquotes\degsmartquotes\degdumbquotes\smartquotes}
244 \renewcommand{\texttt}[1]{\{\ttfamily\addfontfeature{RawFeature={+qtbye,-tlig}}
↪ #1}}
245 }
246
247 %%%%%%%%%%%
248 %%%%%%%%%%%
249 % auxiliary punctuation
250 %%%%%%%%%%%
251 %%%%%%%%%%%
252 % Essentially to be used where the
253 % thing fails to provide the
254 % quotation or
255 % quote like punctuation
256 % needed
257 %%%%%%%%%%%
258 %%%%%%%%%%%
259 %%%%%%%%%%%
260 %%%%%%%%%%%
261 %%%%%%%%%%%
262 %%%%%%%%%%%
263 %%%%%%%%%%%
264 %%%%%%%%%%%
265 % German quotations
266 %%%%%%%%%%%
267 %%%%%%%%%%%
268 \newcommand{\desingle}{\addfontfeature{RawFeature={-qtbye,-tlig}}\symbol{"201A}}
269 \newcommand{\dedouble}{\addfontfeature{RawFeature={-qtbye,-tlig}}\symbol{"201E}}
270 \newcommand{\degmsingle}{\addfontfeature{RawFeature={-qtbye,-tlig}}\symbol{"203A}}
271 \newcommand{\degmdouble}{\addfontfeature{RawFeature={-qtbye,-tlig}}\symbol{"00BB}}
272 %%%%%%%%%%%
273 %%%%%%%%%%%
274 % backtick
275 %%%%%%%%%%%
276 %%%%%%%%%%%

```

```

277 \newcommand{\bcktkc}{\addfontfeature{RawFeature={-qtbye,-tlig}}`}
278 %%%%%%%%%
279 %%%%%%%%%
280 % Straight double
281 % and single quotes
282 %%%%%%%%%
283 %%%%%%%%%
284
285 \newcommand{\dqone}{\addfontfeature{RawFeature={-qtbye,-tlig}}\symbol{"0027}}
286 \newcommand{\dqtwo}{\addfontfeature{RawFeature={-qtbye,-tlig}}\symbol{"0022}}
287 %%%%%%%%%
288 %%%%%%%%%
289 % Prime, mostly for
290 % Feet and inches
291 %%%%%%%%%
292 %%%%%%%%%
293 \newcommand{\lqprime}{'}
294 \newcommand{\lqdoubleprime}{"}
295 %%%%%%%%%
296 %%%%%%%%%
297 % The Okina, for typing
298 % Hawai'i
299 %%%%%%%%%
300 %%%%%%%%%
301 \newcommand{\okina}{'}
302 %%%%%%%%%
303 %%%%%%%%%
304 % The individual smart quotes
305 %
306 %%%%%%%%%
307 %%%%%%%%%
308 %%%%%%%%%
309 %%%%%%%%%
310 \newcommand{\sqtwoleft}{\addfontfeature{RawFeature={-qtbye,-tlig}}"}
311 \newcommand{\sqtwoyright}{\addfontfeature{RawFeature={-qtbye,-tlig}}"}
312 \newcommand{\sqoneright}{\addfontfeature{RawFeature={-qtbye,-tlig}}'}
313 \newcommand{\apost}{\addfontfeature{RawFeature={-qtbye,-tlig}}'}
314 %sqoneleft command moved because order mattered
315 \newcommand{\glmtl}{\addfontfeature{RawFeature={-qtbye,-tlig}}«,}
316 \newcommand{\glmtr}{\addfontfeature{RawFeature={-qtbye,-tlig}}\,»}
317 \newcommand{\sglmtl}{\addfontfeature{RawFeature={-qtbye,-tlig}}<\,}
318 \newcommand{\sglmtr}{\addfontfeature{RawFeature={-qtbye,-tlig}}\,»}
319
320 \newcommand{\degmtl}{\addfontfeature{RawFeature={-qtbye,-tlig}}»}
321 \newcommand{\degmtr}{\addfontfeature{RawFeature={-qtbye,-tlig}}«}
322 \newcommand{\desgmtl}{\addfontfeature{RawFeature={-qtbye,-tlig}}>}
323 \newcommand{\desgmtr}{\addfontfeature{RawFeature={-qtbye,-tlig}}<}
324 %%%%%%%%%

```

```

325 %%%%%%%%%%%
326 % DEUTSCH
327 %%%%%%%%%%%
328 %%%%%%%%%%%
329 \luaexec{function dedoublequotes ( s )
330     return ( s:gsub ( “(.-)” , “,%1“ ) )
331     end}
332 \luaexec{function desinglequotelinestart ( s )
333     return (s:gsub (“^” ,“,” ) )
334     end}
335 \luaexec{function desinglequotesclose( s )
336 return ( s:gsub ( “ ‘(.-)’” , “ ,%1`” ) )
337     end}
338
339 %% Two utility macros to activate/deactivate the Lua function:
340 \newcommand\dedoublequoteson{\directlua{luatexbase.add_to_callback (
341     “process_input_buffer” , dedoublequotes , “dedoublequotes” )}}
342 \newcommand\dedoublequotesoff{\directlua{luatexbase.remove_from_callback (
343     “process_input_buffer” , “dedoublequotes” )}}
344 \newcommand\desinglequotelinestarton{\directlua{luatexbase.add_to_callback (
345     “process_input_buffer” , desinglequotelinestart , “desinglequotelinestart”
346     ↪ )}}
347 \newcommand\desinglequotelinestartoff{\directlua{luatexbase.remove_from_callback
348     ↪ (
349     “process_input_buffer” , “desinglequotelinestart” )}}
350 \newcommand\desinglequotescloseon{\directlua{luatexbase.add_to_callback (
351     “process_input_buffer” , desinglequotesclose , “desinglequotesclose” )}}
352 \newcommand\desinglequotescloseoff{\directlua{luatexbase.remove_from_callback (
353     “process_input_buffer” , “desinglequotesclose” )}}
354 \newcommand{\desmartquotes}{\dedoublequoteson
355 \desinglequotelinestarton
356 \desinglequotescloseon
357 }
358 \newcommand{\dedumbquotes}{\dedoublequotesoff
359 \desinglequotelinestartoff
360 \desinglequotescloseoff
361 \desinglequotespoff
362 \desinglequotespcloseoff
363 }
364 \DeclareOption{de}{
365 \AtBeginDocument{
366 \frsmartquotes
367 \frdumbquotes
368 \smartquotes
369 \dumbquotes\dumbquotes
370 \desmartquotes}

```



```

371 \renewcommand{\texttt}[1]{\ttfamily\addfontfeature{RawFeature={+qtbye,-tlig}}
↪ #1}}
372 }
373
374 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
375 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
376 % SCHÖNERES DEUTSCH
377 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
378 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
379 \luaexec{function degmdoublequotes ( s )
380     return ( s:gsub ( ““(.-)”’ , “»\%1«” ) )
381     end}
382 \luaexec{function degmsinglequotelinstart ( s )
383     return (s:gsub (“^’”,“,”) )
384     end}
385 \luaexec{function degmsinglequotesclose( s )
386     return ( s:gsub ( “ ‘(.-)’” , “ ,\%1«” ) )
387     end}
388
389 %% Two utility macros to activate/deactivate the Lua function:
390 \newcommand\degmdoublequoteson{\directlua{luatexbase.add_to_callback (
391     “process_input_buffer” , degmdoublequotes , “degmdoublequotes” )}}
392 \newcommand\degmdoublequotesoff{\directlua{luatexbase.remove_from_callback (
393     “process_input_buffer” , “degmdoublequotes” )}}
394 \newcommand\degmsinglequotelinstarton{\directlua{luatexbase.add_to_callback (
395     “process_input_buffer” , degmsinglequotelinstart ,
↪ “degmsinglequotelinstart” )}}
396 \newcommand\degmsinglequotelinstartoff{\directlua{luatexbase.remove_from_callback
↪ (
397     “process_input_buffer” , “degmsinglequotelinstart” )}}
398 \newcommand\degmsinglequotescloseon{\directlua{luatexbase.add_to_callback (
399     “process_input_buffer” , degmsinglequotesclose , “degmsinglequotesclose” )}}
400 \newcommand\degmsinglequotescloseoff{\directlua{luatexbase.remove_from_callback
↪ (
401     “process_input_buffer” , “degmsinglequotesclose” )}}
402 \newcommand{\degmsmartquotes}{\degmdoublequoteson
403 \degmsinglequotelinstarton
404 \degmsinglequotescloseon}
405 \newcommand{\degmdumbquotes}{\degmdoublequotesoff
406 \degmsinglequotelinstartoff
407 \degmsinglequotescloseoff}
408 \DeclareOption{degm}{
409 \AtBeginDocument{
410 \frsmartquotes
411 \frdumbquotes
412 \smartquotes
413 \dumbquotes\dumbquotes
414 \degmsmartquotes}

```

```

415 \renewcommand{\texttt}[1][{\ttfamily\addfontfeature{RawFeature={+qtbye,-tlig}}
    ↪ #1}}
416 }
417
418 %%%%%%%%%%%
419 %%%%%%%%%%%
420 % Français
421 %%%%%%%%%%%
422 %%%%%%%%%%%
423 \luaexec{function frdoublequotes ( s )
424     return ( s:gsub ( ““(.-)”” , “«\,\,%1\,\,»” ) )
425     end}
426
427
428 \luaexec{function frsinglequotelinestart ( s )
429     return (s:gsub (“^’”,“'” ) )
430     end}
431 \luaexec{function frsinglequotesclose( s )
432     return ( s:gsub ( “ ‘(.-)’” , “ ‹\,\,%1\,\,›” ) )
433     end}
434
435 %% Two utility macros to activate/deactivate the Lua function:
436 \newcommand\frdoublequoteson{\directlua{luatexbase.add_to_callback (
437     “process_input_buffer” ,frdoublequotes , “frdoublequotes” )}}
438 \newcommand\frdoublequotesoff{\directlua{luatexbase.remove_from_callback (
439     “process_input_buffer” , “frdoublequotes” )}}
440 \newcommand\frsinglequotelinestarton{\directlua{luatexbase.add_to_callback (
441     “process_input_buffer” ,frsinglequotelinestart , “frsinglequotelinestart”
    ↪ )}}
442 \newcommand\frsinglequotelinestartoff{\directlua{luatexbase.remove_from_callback
    ↪ (
443     “process_input_buffer” , “frsinglequotelinestart” )}}
444 \newcommand\frsinglequotescloseon{\directlua{luatexbase.add_to_callback (
445     “process_input_buffer” ,frsinglequotesclose , “frsinglequotesclose” )}}
446 \newcommand\frsinglequotescloseoff{\directlua{luatexbase.remove_from_callback (
447     “process_input_buffer” , “frsinglequotesclose” )}}
448 \newcommand{\frsmartquotes}{\frdoublequoteson
449 \frsinglequotespon
450 \frsinglequotespcloseon
451 \frsinglequotelinestarton
452 \frsinglequotescloseon}
453 \newcommand{\frdumbquotes}{\frdoublequotesoff
454 \frsinglequotespoff
455 \frsinglequotespcloseoff
456 \frsinglequotelinestartoff
457 \frsinglequotescloseoff}
458 \DeclareOption{fr}{
459 \AtBeginDocument{\desmartquotes\dedumbquotes%

```

```

460 \degsmartquotes\degmdumbquotes%
461 \smartquotes\dumbquotes%
462 \dumbquotes\frsmartquotes}
463 \renewcommand{\texttt}[1]{\ttfamily\addfontfeature{RawFeature={+qtbye,-tlig}}
↪ #1}}
464
465
466 }
467
468 %%%%%%%%%%%
469 % Default option is English
470 %%%%%%%%%%%
471 %%%%%%%%%%%
472 \ExecuteOptions{en}%
473 \ProcessOptions*
474
475
476
477

```

7 Version History

1.3.0

15 January 2023 fixed single quotes and parentheses

1.2.2

Added the “degm” option

1.2.1

Suspended automatic elision support due to implementation issues

1.1.0

25 October 2022: Added support for elision

1.0.1

30 August 2022: Further corrections to bug preventing non-English usage

1.0.1

28 August 2022: Change bug preventing non-English usage

1.0.0

21 August 2022: Package creation