

# Documented Code For mfirstuc v2.05

Nicola L.C. Talbot

Dickimaw Books

<http://www.dickimaw-books.com/>

2017-11-10

## Contents

<b>1 Introduction</b>	<b>1</b>
<b>2 Main Package Code</b>	<b>1</b>
2.1 Mfirstuc Documented Code . . . . .	1
2.2 Mfirstuc-english Documented Code . . . . .	12
<b>Change History</b>	<b>13</b>
<b>Change History</b>	<b>13</b>
<b>Index</b>	<b>14</b>

## 1 Introduction

This is the documented code for the mfirstuc package. See mfirstuc-manual.pdf for the main user guide.

## 2 Main Package Code

### 2.1 Mfirstuc Documented Code

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{mfirstuc}[2017/11/10 v2.05 (NLCT)]
```

Requires etoolbox:

```
3 \RequirePackage{etoolbox}
```

\makefirstuc Syntax:

```
\makefirstuc{<text>}
```

Makes the first letter uppercase, but will skip initial control sequences if they are followed by a group and make the first thing in the group uppercase, unless the group is empty. Thus `\makefirstuc{abc}` will produce: `Abc`, `\makefirstuc{\ae bc}` will produce: `Æbc`, but `\makefirstuc{\emph{abc}}` will produce `Abc`. This is required by `\Gls` and `\Glspl`.

```

4 \newif\if@glscs
5 \newtoks\@glsmfirst
6 \newtoks\@glsmrest
7 \newrobustcmd*{\makefirstuc}[1]{%

```

Check if argument starts with `\protect` which needs to be skipped.

```

8 \@MFU@p@makefirstuc#1\relax\relax\relax\@end@MFU@protected{#1}%
9 }

```

`\@MFU@protect`

```

10 \newcommand*{\@MFU@protect}{\protect}

```

`\@p@makefirstuc`

Internal command to do the required action.

```

11 \def\@MFU@p@makefirstuc#1#2#3\@end@MFU@protected#4{%

```

Test if the first argument is `\protect`.

```

12 \def\@glsc@argi{#1}%
13 \ifx\@glsc@argi\@MFU@protect
14 \@makefirstuc{#2#3}%
15 \else
16 \@makefirstuc{#4}%
17 \fi
18 }

```

`\@makefirstuc`

Internal command to do the required action.

```

19 \newcommand*{\@makefirstuc}[1]{%
20 \def\@glsc@argi{#1}%
21 \ifx\@glsc@argi\@empty

```

If the argument is empty, do nothing.

```

22 \else

23 \def\@glsc@tmp{\ #1}%
24 \@onelevel@sanitize\@glsc@tmp
25 \expandafter\@glsc@checkcs\@glsc@tmp\relax\relax
26 \if@glscs

```

Check if we have two sequential control sequences.

```

27 \expandafter\def\expandafter\@glsc@tmp\expandafter{\expandafter
28 \ \@gobble #1}%
29 \@onelevel@sanitize\@glsc@tmp
30 \expandafter\@glsc@checkcs\@glsc@tmp\relax\relax
31 \if@glscs\@glscsfalse\else\@glscstrue\fi
32 \fi
33 \if@glscs
34 \@glsc@getbody #1{}\@nil

```

```

35     \ifx\@gls@rest\@empty
36     \glsmakefirstuc{#1}%
37   \else
38     \expandafter\@gls@split\@gls@rest\@nil
39     \ifx\@gls@first\@empty
40     \glsmakefirstuc{#1}%
41   \else
42     \expandafter\@glsmfirst\expandafter{\@gls@first}%
43     \expandafter\@glsmrest\expandafter{\@gls@rest}%
44     \edef\@gls@domfirstuc{\noexpand\@gls@body
45       {\noexpand\glsmakefirstuc\the\@glsmfirst}%
46       \the\@glsmrest}%
47     \@gls@domfirstuc
48   \fi
49 \fi
50 \else
51   \glsmakefirstuc{#1}%
52 \fi
53 \fi
54 }

```

`\@gls@split` Put first argument in `\@gls@first` and second argument in `\@gls@rest`:

```

55 \ifdef\dtl@getfirst@UTFviii
56 {%
57   \def\@gls@split#1#2\@nil{%
58     \ifbool{\dtl@utf8}
59     {%
60       \expandafter\dtl@if@two@octets#1#2\relax\dtl@end@if@two@octets
61     {%
62       \dtl@getfirst@UTFviii#1#2\@nil\end\dtl@getfirst@UTFviii
63       \let\@gls@first\dtl@first
64       \let\@gls@rest\dtl@rest
65     }%
66   {%
67     \def\@gls@first{#1}\def\@gls@rest{#2}%
68   }%
69 }%
70 {%
71   \def\@gls@first{#1}\def\@gls@rest{#2}%
72 }%
73 }
74 }%
75 {%
76   \def\@gls@split#1#2\@nil{%
77     \def\@gls@first{#1}\def\@gls@rest{#2}%
78   }
79 }

80 \def\@gls@checkcs#1 #2#3\relax{%

```

```

81 \def\@gls@argi{#1}\def\@gls@argii{#2}%
82 \ifx\@gls@argi\@gls@argii
83   \@glscstrue
84 \else
85   \@glscsfalse
86 \fi
87 }

```

Has datatool-base v2.24 (or above) been loaded? If so and we have UTF-8 enabled, we can grab the first octet.

fu@applytofirst

```

88 \ifdef\dtl@getfirst@UTFviii
89 {%
90   \def\@mfu@applytofirst#1#2\@mfu@end@applytofirst#3{%
91     \expandafter\dtl@if@two@octets#1#2\relax\dtl@end@if@two@octets
92     {%
93       \dtl@getfirst@UTFviii#1#2\@nil\end\dtl@getfirst@UTFviii
94       \expandafter#3\expandafter{\dtl@first}\dtl@rest
95     }%
96   {%

```

First argument needs to be grouped in case it consists of more than one token.

```

97     #3{#1}#2%
98   }%
99 }%
100 }%
101 {%

```

First argument needs to be grouped in case it consists of more than one token.

```

102 \def\@mfu@applytofirst#1#2\@mfu@end@applytofirst#3{#3{#1}#2}
103 }

```

\MFUapplytofirst

\MFUapplytofirst{<cs>}{<text>}

Apply <cs> to first token of text, or first UTF-8 character if datatool-base v2.24 above has been loaded with the utf8 support enabled.

```

104 \ifdef\dtl@getfirst@UTFviii
105 {%
106   \newcommand*\MFUapplytofirst}[2]{%
107     \ifbool{@dtl@utf8}
108     {%
109       \ifblank{#2}%
110       {#1{#2}}%
111     }%
112     \ifblank{#1}%
113     {#1#2}%
114     {%
115       \@mfu@applytofirst#2\@mfu@end@applytofirst#1\relax

```

```

116         }%
117     }%
118 }%
119 {%
120     #1#2%
121 }%
122 }
123 }%
124 {%
125     \newcommand*{\MFUapplytofirst}[2]{#1#2}
126 }

```

`\gls@makefirstuc` Make first thing upper case:

```
127 \def\@gls@makefirstuc#1{\MFUapplytofirst\mfirstucMakeUppercase{#1}}
```

`\ucMakeUppercase` Allow user to replace `\MakeUppercase` with another case changing command.

```
128 \newcommand*{\mfirstucMakeUppercase}{\MakeUppercase}
```

`\glsmakefirstuc` Provide a user command to make it easier to customise.

```
129 \newcommand*{\glsmakefirstuc}[1]{\@gls@makefirstuc{#1}}
```

Get the first grouped argument and store in `\@gls@body`.

```
130 \def\@gls@getbody#1#{\def\@gls@body{#1}\@gls@gobbletonil}
```

Scoup up everything to `\@nil` and store in `\@gls@rest`:

```
131 \def\@gls@gobbletonil#1\@nil{\def\@gls@rest{#1}}
```

`\xmakefirstuc` Expand argument once before applying `\makefirstuc` (added v1.01).

```

132 \newcommand*{\xmakefirstuc}[1]{%
133     \expandafter\makefirstuc\expandafter{#1}%
134 }

```

`\emakefirstuc` Fully expand argument before applying `\makefirstuc`

```

135 \newrobustcmd*{\emakefirstuc}[1]{%
136     \protected@edef\@MFU@caparg{#1}%
137     \expandafter\makefirstuc\expandafter{\@MFU@caparg}%
138 }

```

`\capitalisewords` Capitalise each word in the argument. Words are considered to be separated by plain spaces (i.e. non-breakable spaces won't be considered a word break).

```

139 \newrobustcmd*{\capitalisewords}[1]{%
140     \let\@mfu@checkword\@gobble
141     \def\gls@add@space{}%
142     \let\@mfu@domakefirstuc\MFUcapword
143     \mfu@capitalisewords#1 \@nil\mfu@endcap
144 }

```

capitalisewords \@mfu@checkword needs to be set before use.

```

145 \newcommand*{\@mfu@mid@capitalisewords}[1]{%
146   \def\gls@add@space{}%
147   \let\@mfu@domakefirstuc\MFUcapword
148   \mfu@capitalisewords#1 \@nil\mfu@endcap
149 }

150 \def\mfu@capitalisewords#1 #2\mfu@endcap{%
151   \def\mfu@cap@first{#1}%
152   \def\mfu@cap@second{#2}%
153   \gls@add@space
154   \@mfu@checkword{#1}%
155   \@mfu@domakefirstuc{#1}%
156   \def\gls@add@space{ }%
157   \ifx\mfu@cap@second\@nnil
158     \let\next@mfu@cap\mfu@noop
159   \else
160     \let\next@mfu@cap\mfu@capitalisewords
161     \let\@mfu@checkword\mfu@checkword
162   \fi
163   \next@mfu@cap#2\mfu@endcap
164 }
165 \def\mfu@noop#1\mfu@endcap{}

```

\ifMFUhyphen Split on hyphens.

```

166 \newif\ifMFUhyphen
167 \MFUhyphenfalse

```

\MFUcapword

```

168 \newcommand*{\MFUcapword}[1]{%
169   \ifMFUhyphen
170     \@MFUcapword#1-\@nil\@endMFUcapword
171   \else
172     \makefirstuc{#1}%
173   \fi
174 }

```

\@MFUcapword

```

175 \def\@MFUcapword#1-#2\@endMFUcapword{%
176   \makefirstuc{#1}%
177   \def\@mfu@argii{#2}%
178   \ifx\@mfu@argii\@nnil
179   \else
180     -\@MFUcapword#2\@endMFUcapword
181   \fi
182 }

```

\mfu@checkword Check if word should be capitalised. This originally used etoolbox's \ifinlist command but this doesn't work if the word contains grouping.

```

183 \newcommand*\mfu@checkword[1]{%
184   \def\mfu@checkword@arg{#1}%
185   \let\@mfu@domakefirstuc\MFUcapword
186   \forlistloop\mfu@checkword@do\@mfu@nocaplist
187 }

mfu@checkword@do  Handler for loop used by \mfu@checkword
188 \newcommand*\mfu@checkword@do[1]{%
189   \def\mfu@checkword@argii{#1}%
190   \ifdefequal\mfu@checkword@arg\mfu@checkword@argii
191   {%
192     \let\@mfu@domakefirstuc\@firstofone
193     \listbreak
194   }%
195   {%

Need to check if \@mfu@terminator has been tacked onto the end.
196     \def\mfu@checkword@argii{#1{\@mfu@terminator}}%
197     \ifdefequal\mfu@checkword@arg\mfu@checkword@argii
198     {%
199       \let\@mfu@domakefirstuc\@firstofone
200       \listbreak
201     }%
202     {}%
203   }%
204 }

\@mfu@nocaplist  List of words that shouldn't be capitalised.
205 \newcommand*\@mfu@nocaplist{}}

\MFUUnocap  Provide the user with a means to add a word to the list. (No expansion performed on the
argument.)
206 \newcommand*\MFUUnocap[1]{\listadd{\@mfu@nocaplist}{#1}}

\gMFUUnocap  Global version.
207 \newcommand*\gMFUUnocap[1]{\listgadd{\@mfu@nocaplist}{#1}}

\MFUClear  Clear the list
208 \newcommand*\MFUClear{\renewcommand*\@mfu@nocaplist{}}

capitalisewords  Short-cut command:
209 \newcommand*\xcapitalisewords[1]{%
210   \expandafter\capitalisewords\expandafter{#1}%
211 }

capitalisewords  Fully expand argument before applying \capitalisewords
212 \newrobustcmd*\ecapitalisewords[1]{%
213   \protected@edef\@MFU@caparg{#1}%
214   \expandafter\capitalisewords\expandafter{\@MFU@caparg}%
215 }

```

```

\@mfu@nil
216 \newcommand\@mfu@nil{\unskip}

\@mfu@nnil
217 \newcommand*\@mfu@nnil{\@mfu@nil}

italisefmtwords Like \capitalisewords but assumes that the phrase contains formatting text-block com-
mands.
218 \newcommand*{\capitalisefmtwords}{%
219 \ifstar\@s@capitalisefmtwords\@ns@capitalisefmtwords
220 }

italisefmtwords Starred form.
221 \newcommand*{\@s@capitalisefmtwords}[1]{%
222 \@s@checkgrp@capitalisefmtwords#1\@nil
223 }

italisefmtwords
224 \newcommand*{\@s@checkgrp@capitalisefmtwords}{%
225 \ifnextchar\bgroup{\@s@noc@capitalisefmtwords}{\@s@capitalisefmtwords}%
226 }

italisefmtwords Argument starts with a group.
227 \def\@s@noc@capitalisefmtwords#1\@nil{%
228 \capitalisewords{#1}%
229 }

italisefmtwords
230 \newcommand*{\@s@capitalisefmtwords}[1]{%
231 \def\gls@argi{#1}%
232 \ifx\gls@argi\@MFU@protect
Skip \protect
233 \let\@mfu@capfmtwordsnext\@s@capitalisefmtwords
234 \else
235 \ifx\gls@argi\@nnil
Nothing to do.
236 \let\@mfu@capfmtwordsnext\relax
237 \else
238 \def\@gls@tmp{\ #1}%
239 \@onelevel@sanitize\@gls@tmp
240 \expandafter\@gls@checkcs\@gls@tmp\relax\relax
241 \if@gls@cs
242 \def\@mfu@capfmtwordsnext{\@s@cs@capitalisefmtwords#1}%
243 \else
244 \def\@mfu@capfmtwordsnext{\@s@noc@capitalisefmtwords#1}%
245 \fi
246 \fi

```



```

247 \fi
248 \@mfu@capfmtwordsnext
249 }

italisefmtwords
250 \newcommand*{\@cs@capitalisefmtwords}[1]{%
251   \@ifnextchar\bgroup
252     {\@cs@cs@g@capitalisefmtwords#1}%
253     {\@cs@cs@ng@capitalisefmtwords#1}%
254 }

italisefmtwords
255 \def\@cs@cs@g@capitalisefmtwords#1#2#3\@nil{%
256   #1{\capitalisewords{#2}}%
257   \ifblank{#3}%
258   {}%
259   {%
260     \def\gls@add@space{}%
261     \let\mfu@checkword\mfu@checkword
262     \mfu@capitalisewords#3 \@nil\mfu@endcap
263   }%
264 }

italisefmtwords
265 \def\@cs@cs@ng@capitalisefmtwords#1#2\@nil{%
266   \capitalisewords{#1#2}%
267 }

italisefmtwords  Unstarred form.
268 \newcommand*{\@ns@capitalisefmtwords}[1]{%
269   \let\mfu@checkword\gobble
270   \def\gls@add@space{}%
271   \@capitalisefmtwords#1 \@mfu@nil{} \unskip
272 }

italisefmtwords  Take care of group at the start of the word.
273 \newcommand*{\@capitalisefmtwords}{%
274   \@ifnextchar\bgroup{\@capitalise@g@fmtwords}{\@capitalise@ng@fmtwords}%
275 }
276 \newcommand*{\@capitalise@g@fmtwords}[1]{%
277   \@capitalise@ng@fmtwords{{#1}}%
278 }
279 \def\mfu@ifnil#1#2\mfu@end@nil#3#4{%
280   \def\gls@argi{#1}%
281   \ifx\gls@argi\mfu@nnil
282     #3%
283   \else
284     #4%
285   \fi

```

```

286 }
287 \def\@capitalise@eng@fmtwords#1 {%
288   \mfu@ifnil#1\relax\relax\mfu@end@nil
289   {%
290     \def\mfu@capfmtwordsnext{#1}%
291   }%
292   {%
293     \def\mfu@capfmtwordsnext{%
294       \@capitalise@fmtwords#1\mfu@nil
295       \def\gls@add@space{ }%
296       \let\mfu@checkword\mfu@checkword
297       \@cap@nofmtphrase}%
298   }%
299   \mfu@capfmtwordsnext
300 }%

```

talise@fmtwords

```

301 \def\@capitalise@fmtwords#1{%
302   \mfu@ifnil#1\relax\relax\mfu@end@nil
303   {%
304     Do nothing.
305     \def\mfu@capfmtwordsnext{#1}%
306   }%
307   {%
308     \def\gls@argi{#1}%
309     \ifx\gls@argi\@MFU@protect
310       Skip \protect
311       \def\mfu@capfmtwordsnext{\@capitalise@fmtwords}%
312     \else
313       \def\@gls@tmp{\ #1}%
314       \@onelevel@sanitize\@gls@tmp
315       \expandafter\@gls@checkcs\@gls@tmp\relax\relax
316       \if@glscs
317         \def\mfu@capfmtwordsnext{\@capitalisefmtwords#1}%
318       \else
319         \ifx\gls@argi\@empty
320           \def\mfu@capfmtwordsnext{\@capnofmtphrase{}}%
321         \else
322           \def\mfu@capfmtwordsnext{\@capnofmtphrase#1}%
323         \fi
324       \fi
325     }%
326   }%
327   \mfu@capfmtwordsnext
328 }%

```

italisefmtwords Check for a group following argument.

```

327 \def\@@capitalisefmtwords#1{%

```

```

328 \@ifnextchar\bgroup{\@@capfmtsubphrase#1}{\@@capnofmtphrase#1}%
329 }

@capnofmtphrase
330 \def\@@capnofmtphrase{%
331 \ifnextchar\bgroup{\@@cap@g@nofmtphrase}{\@@cap@ng@nofmtphrase}%
332 }

@ng@nofmtphrase
333 \def\@@cap@ng@nofmtphrase#1\@mfu@nil{%
334 \gls@add@space
335 \@mfu@mid@capitalisewords{#1}%
336 \def\gls@add@space{ }%
337 }

p@g@nofmtphrase
338 \def\@@cap@g@nofmtphrase#1{%
339 \@@cap@ng@nofmtphrase{#1}%
340 }

@mfu@terminator
341 \newcommand*\@mfu@terminator{\hbox{}}

capfmtsubphrase
342 \def\@@capfmtsubphrase#1#2#3{%
    Protect any trailing space at the end of the second argument.
343 #1{\@capitalisefmtwords#2{\@mfu@terminator} \@mfu@nil{} \unskip}%
344 \def\gls@add@space{ }%
345 \let\@mfu@checkword\mfu@checkword
346 \@mfu@ifnil#1\relax\relax\@mfu@end@nil
347 {\let\@mfu@fmtsubphrasenext\relax}%
348 {\def\@mfu@fmtsubphrasenext{#3}}%
349 \@mfu@fmtsubphrasenext
350 }

cap@nofmtphrase
351 \def\@@cap@nofmtphrase#1 {%
352 \@mfu@ifnil#1\relax\relax\@mfu@end@nil
353 {%
354 \def\@mfu@capnofmtwordsnext{#1 }%
355 }%
356 {%
357 \def\@mfu@capnofmtwordsnext{%
358 \@capitalisefmtwords#1 \@mfu@nil{} \unskip
359 \def\gls@add@space{ }%
360 \let\@mfu@checkword\mfu@checkword
361 \@capitalisefmtwords
362 }%

```

```

363 }%
364 \@mfu@capnofmtwordsnext
365 }

```

`italisefmtwords` Short-cut command:

```

366 \newcommand*{\xcapitalisefmtwords}{%
367   \@ifstar\@s@xcapitalisefmtwords\@ns@xcapitalisefmtwords
368 }

```

`italisefmtwords` Starred version

```

369 \newcommand*{\@s@xcapitalisefmtwords}[1]{%
370   \expandafter\@s@capitalisefmtwords\expandafter{#1}%
371 }

```

`italisefmtwords` Unstarred version

```

372 \newcommand*{\@ns@xcapitalisefmtwords}[1]{%
373   \expandafter\@ns@capitalisefmtwords\expandafter{#1}%
374 }

```

`italisefmtwords`

```

375 \newrobustcmd*{\ecapitalisefmtwords}{%
376   \@ifstar\@s@ecapitalisefmtwords\@ns@ecapitalisefmtwords
377 }

```

`italisefmtwords` Starred version

```

378 \newcommand*{\@s@ecapitalisefmtwords}[1]{%
379   \protected@edef\@MFU@caparg{#1}%
380   \expandafter\@s@capitalisefmtwords\expandafter{\@MFU@caparg}%
381 }

```

`italisefmtwords` Unstarred version

```

382 \newcommand*{\@ns@ecapitalisefmtwords}[1]{%
383   \protected@edef\@MFU@caparg{#1}%
384   \expandafter\@ns@capitalisefmtwords\expandafter{\@MFU@caparg}%
385 }

```

## 2.2 Mfirstuc-english Documented Code

```

386 \NeedsTeXFormat{LaTeX2e}
387 \ProvidesPackage{mfirstuc-english}[2017/11/10 v2.05 (NLCT)]

```

Load mfirstuc if not already loaded:

```

388 \RequirePackage{mfirstuc}

```

Add no-cap words. This list isn't complete. I don't intend adding disputed words. Additional words can be added to the document in a similar manner.

```

389 \MFUnocap{a}
390 \MFUnocap{an}
391 \MFUnocap{and}

```

392 \MFUnocap{but}  
 393 \MFUnocap{for}  
 394 \MFUnocap{in}  
 395 \MFUnocap{of}  
 396 \MFUnocap{or}  
 397 \MFUnocap{no}  
 398 \MFUnocap{nor}  
 399 \MFUnocap{so}  
 400 \MFUnocap{some}  
 401 \MFUnocap{the}  
 402 \MFUnocap{with}  
 403 \MFUnocap{yet}

## Change History

1.03 (2008/12/22)	\mfu@checkword@do: new ..... 7
\@makefirstuc: changed	2.03 (2016-07-29)
\protected@edef to \def ..... 2	\@@cap@nofmtphrase: new ..... 11
1.05 (2011/04/02)	\@@capfmtsubphrase: new ..... 11
\glsmakefirstuc: new ..... 5	\@@capitalisefmtwords: new ..... 10
1.06 (2012/05/21)	\@@capnofmtphrase: new ..... 11
General: now requires etoolbox ..... 1	\@MFUp@makefirstuc: changed test .... 2
\capitalisewords: new ..... 5	\@MFUp@protect: new ..... 2
\xcapitalisewords: new ..... 7	\@MFUcapword: new ..... 6
1.08 (2013-11-04)	\@capitalise@fmtwords: new ..... 10
\capitalisewords: made robust ..... 5	\@capitalisefmtwords: new ..... 9
\makefirstuc: made robust ..... 2	\@gls@split: added check for
1.09 (2017-07-30)	datatool-base UTF-8 support ..... 3
\@mfu@nocaplist: new ..... 7	\@mfu@applytofirst: new ..... 4
\capitalisewords: added check for	\@mfu@mid@capitalisewords: new .... 6
words that shouldn't be capitalised ... 5	\capitalisefmtwords: new ..... 8
\gMFUnocap: new ..... 7	\ecapitalisefmtwords: new ..... 12
\mfu@checkword: new ..... 6	\ifMFUhyphen: new ..... 6
\MFUclear: new ..... 7	\MFUapplytofirst: new ..... 4
1.10 (2015/05/03)	\MFUcapword: new ..... 6
\ecapitalisewords: new ..... 7	\xcapitalisefmtwords: new ..... 12
\emakefirstuc: new ..... 5	2.04 (2016-07-31)
2.0 (2015/09/09)	\@@cap@g@nofmtphrase: new ..... 11
\@makefirstuc: added check for two	\@@cap@ng@nofmtphrase: new ..... 11
control sequences ..... 2	\@@s@capitalisefmtwords: new ..... 8
General: package split from glossaries .... 1	\@@s@checkgrp@capitalisefmtwords:
2.01 (2015-12-08)	new ..... 8
\makefirstuc: added check for	\@@s@cs@capitalisefmtwords: new ... 9
\protect ..... 2	\@@s@cs@g@capitalisefmtwords: new . 9
2.02 (2015-12-17)	\@@s@cs@ng@capitalisefmtwords:
\mfu@checkword: modified to use loop	new ..... 9
instead of \ifinlist ..... 6	\@@s@nocs@capitalisefmtwords: new . 8

\@gls@split: added missing false part .. 3	\@s@ecapitalisefmtwords: new ..... 12
\@mfu@terminator: new ..... 11	\@s@xcapitalisefmtwords: new ..... 12
\@ns@capitalisefmtwords: new ..... 9	2.05 (2017-11-10)
\@ns@ecapitalisefmtwords: new .... 12	\@mfu@applytofirst: grouped first
\@ns@xcapitalisefmtwords: new .... 12	argument ..... 4
\@s@capitalisefmtwords: new ..... 8	

## Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	
\@@cap@g@nofmtphrase ..... 11	\@gls@split ..... 3
\@@cap@ng@nofmtphrase ..... 11	\@gls@tmp ..... 2, 8, 10
\@@cap@nofmtphrase ..... 10	\@glscsfalse ..... 2, 4
\@@capfmtsubphrase ..... 11	\@glscstrue ..... 2, 4
\@@capitalisefmtwords ..... 10	\@glsmfirst ..... 2, 3
\@@capnofmtphrase ..... 10, 11	\@glsmrest ..... 2, 3
\@s@capitalisefmtwords ..... 8	\@gobble ..... 2, 5, 9
\@s@checkgrp@capitalisefmtwords ... 8	\@ifnextchar ..... 8, 9, 11
\@s@cs@capitalisefmtwords ..... 8	\@ifstar ..... 8, 12
\@s@cs@g@capitalisefmtwords ..... 9	\@makefirstuc ..... 2
\@s@cs@ng@capitalisefmtwords ..... 9	\@mfu@applytofirst ..... 4
\@s@nocs@capitalisefmtwords ..... 8	\@mfu@argii ..... 6
\@MFU@caparg ..... 5, 7, 12	\@mfu@capfmtwordsnext ..... 8–10
\@MFU@p@makefirstuc ..... 2	\@mfu@capnofmtwordsnext ..... 11, 12
\@MFU@protect ..... 2, 8, 10	\@mfu@checkword ..... 5, 6, 9–11
\@MFU@capword ..... 6	\@mfu@domakefirstuc ..... 5–7
\@capitalise@fmtwords ..... 10	\@mfu@end@applytofirst ..... 4
\@capitalise@g@fmtwords ..... 9	\@mfu@end@nil ..... 9–11
\@capitalise@ng@fmtwords ..... 9, 10	\@mfu@fmtsubphrasenext ..... 11
\@capitalisefmtwords ..... 9, 11	\@mfu@ifnil ..... 9–11
\@empty ..... 2, 3, 10	\@mfu@mid@capitalisewords ..... 11
\@end@MFU@protected ..... 2	\@mfu@nil ..... 8–11
\@end@MFU@capword ..... 6	\@mfu@nnil ..... 9
\@firstofone ..... 7	\@mfu@nocaplist ..... 7
\@gls@argi ..... 4	\@mfu@terminator ..... 7, 11
\@gls@argii ..... 4	\@nil ..... 2–6, 8, 9
\@gls@body ..... 3, 5	\@nnil ..... 6, 8
\@gls@checkcs ..... 2, 3, 8, 10	\@ns@capitalisefmtwords ..... 8, 12
\@gls@domfirstuc ..... 3	\@ns@ecapitalisefmtwords ..... 12
\@gls@first ..... 3	\@ns@xcapitalisefmtwords ..... 12
\@gls@getbody ..... 2, 5	\@onelevel@sanitize ..... 2, 8, 10
\@gls@gobbletonil ..... 5	\@s@capitalisefmtwords ..... 8, 12
\@gls@makefirstuc ..... 5	\@s@ecapitalisefmtwords ..... 12
\@gls@rest ..... 3, 5	\@s@xcapitalisefmtwords ..... 12

\_	2, 8, 10	\listadd	7
		\listbreak	7
		\listgadd	7
<b>B</b>		<b>M</b>	
\bgroup	8, 9, 11	\makefirstuc	5, 6
<b>C</b>		\MakeUppercase	5
\capitalisewords	7–9	mfirstuc package	12
<b>D</b>		\mfirstucMakeUppercase	5
datatool-base package	4	\mfu@cap@first	6
\def	2–11	\mfu@cap@second	6
\dtl@end@if@two@octets	3, 4	\mfu@capitalisewords	5, 6, 9
\dtl@first	3, 4	\mfu@checkword	6, 9–11
\dtl@getfirst@UTFviii	3, 4	\mfu@checkword@arg	7
\dtl@if@two@octets	3, 4	\mfu@checkword@argii	7
\dtl@rest	3, 4	\mfu@checkword@do	7
<b>E</b>		\mfu@endcap	5, 6, 9
\edef	3	\mfu@noop	6
\else	2–4, 6, 8–10	\MFUapplytofirst	5
\end@dtl@getfirst@UTFviii	3, 4	\MFUcapword	5–7
etoolbox package	1, 6	\MFUhyphenfalse	6
\expandafter	2–5, 7, 8, 10, 12	\MFUnocap	12, 13
<b>F</b>		<b>N</b>	
\fi	2–4, 6, 8–10	\NeedsTeXFormat	1, 12
\forlistloop	7	\newcommand	2, 4–9, 11, 12
<b>G</b>		\newif	2, 6
\gls@add@space	5, 6, 9–11	\newrobustcmd	2, 5, 7, 12
\gls@argi	2, 8–10	\newtoks	2
\glsmakefirstuc	3	\next@mfu@cap	6
<b>H</b>		\noexpand	3
\hbox	11	<b>P</b>	
<b>I</b>		\protect	2
\if@glscs	2, 8, 10	\protected@edef	5, 7, 12
\ifblank	4, 9	\ProvidesPackage	1, 12
\ifbool	3, 4	<b>R</b>	
\ifdef	3, 4	\relax	2–4, 8, 10, 11
\ifdefequal	7	\renewcommand	7
\ifMFUhyphen	6	\RequirePackage	1, 12
\ifx	2–4, 6, 8–10	<b>T</b>	
<b>L</b>		\the	3
\let	3, 5–11	<b>U</b>	
		\unskip	8, 9, 11