Documented Code For mfirstuc v2.08

Nicola L.C. Talbot

Dickimaw Books

2022-10-14

Contents

1 Introduction

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

2 Main Package Code

3 Mfirstuc Documented Code

```
\NeedsTeXFormat{LaTeX2e}
```

Rollback releases:
```
\DeclareRelease{v2.07}{2021-10-15}{mfirstuc-2021-10-15.sty}
\DeclareCurrentRelease{v2.08}{2022-10-14}
```

Declare package:
```
\ProvidesPackage{mfirstuc}[2022/10/14 v2.08 (NLCT)]
```

Requires etoolbox:
```
\RequirePackage{etoolbox}
```

Version 2.07 has switched to LaTeX3 for the actual case-changing. The original \makefirstuc has been reimplemented to use LaTeX3 code. There is also a new command \MFUsentencecase which just uses \text_titlecase_first:n. Note that \MFUsentencecase is internally used by \glsmakefirstuc, which in turn is used by \makefirstuc to perform the actual case-change.

\glsmakefirstuc Provide a user command to make it easier to customise. This is used internally by \makefirstuc. (The original commands provided by mfirstuc were part of the glossaries package, hence the "gls" prefix.)

```
\newcommand*{\glsmakefirstuc}[1]{\MFUsentencecase{\unexpanded{#1}}}
```

\mfirstucMakeUppercase This was originally provided to allow user to replace \MakeUppercase with another case-changing command. This is now only used with \mfugrabfirstuc.

```
\ExplSyntaxOn
\newcommand*{\mfirstucMakeUppercase}[1]{\text_uppercase:n{#1}}
```

expanded Allow argument to expand. This also resets the definition of \mfirstucMakeUppercase in case it was changed by grabfirst.

```
\DeclareOption{expanded}{%
 \renewcommand*{\glsmakefirstuc}[1]{\MFUsentencecase{#1}}%
 \renewcommand*{\mfirstucMakeUppercase}[1]{\text_uppercase:n{#1}}%
}
```

unexpanded Prevent argument of \glsmakefirstuc from expanding. This also resets the definition of \mfirstucMakeUppercase in case it was changed by grabfirst, but note that it doesn't use \unexpanded.

```
\DeclareOption{unexpanded}{%
 \renewcommand*{\glsmakefirstuc}[1]{\MFUsentencecase{\unexpanded{#1}}}%
 \renewcommand*{\mfirstucMakeUppercase}[1]{\text_uppercase:n{#1}}%
}
\ExplSyntaxOff
```

\mfugrabfirstuc Used by the grabfirst option to emulate v2.07.

```
\newrobustcmd*{\mfugrabfirstuc}[1]{\@mfu@grabfirstuc#1\empty}
```

\@mfu@grabfirstuc

```
\newcommand*{\@mfu@grabfirstuc}[2]{\mfirstucMakeUppercase{#1}#2}
```

grabfirst Emulate v2.07 method.

```
\DeclareOption{grabfirst}{%
 \renewcommand*{\glsmakefirstuc}[1]{\mfugrabfirstuc{#1}}%
 \renewcommand*{\mfirstucMakeUppercase}[1]{\MakeUppercase{#1}}%
}
```

Process options.

```
\ProcessOptions
```

There are differences between \makefirstuc and \MFUsentencecase, the principle ones being expansion occurs differently and \makefirstuc stops at exclusions whereas \MFUsentencecase skips them.

```
\ExplSyntaxOn
```

\MFUsentencecase New command provided that simply uses LaTeX3, which expands the content. This may be preferred over \makefirstuc depending on the content.

```
\newcommand{\MFUsentencecase}[1]{\text_titlecase_first:n{#1}}
```

\MFUexcl Add a command to the exclusion list. This is used by \text_titlecase_first:n to skip commands that shouldn't have their argument changed (such as \ref). It's also used by \makefirstuc.

```
\newcommand{\MFUexcl}[1]{
  \tl_if_empty:nF { #1 }
  {
    \tl_if_in:NnF \l_text_case_exclude_arg_tl {#1}
     {
       \tl_put_right:Nn \l_text_case_exclude_arg_tl {#1}
     }
  }
 }
```

Add \ensuremath as an exclusion otherwise \makefirstuc will try passing its argument to \MFUsentencecase, which will fail.

```
\MFUexcl{\ensuremath}
```

Test if the given command is in the exclusion list.

```
\cs_new:Npn \mfirstuc_if_excluded:NTF #1 #2 #3
 {
  \tl_if_in:NnTF \l_text_case_exclude_arg_tl {#1} {#2} {#3}
 }
```

Provide an implementation of \makefirstuc that matches older versions as much as possible.

```
\tl_new:N \l__mfirstuc_stack_tl
\tl_new:N \l__mfirstuc_head_tl
\tl_new:N \l__mfirstuc_next_tl
```

In addition to exclusions, also provide blockers. These are only supported by \makefirstuc. Exclusions and their argument are skipped, whereas blockers prevent case-change. For example, sentence-casing \label{ex}some text should skip \label{ex}, whereas \nameref{ex} some text shouldn't have any case change applied.

```
\tl_new:N \l__mfirstuc_blocker_tl
```

\MFUblocker Add a command to the blocker list.

```
\newcommand{\MFUblocker}[1]{
 \tl_if_empty:nF { #1 }
  {
```

```
      \tl_if_in:NnF \l__mfirstuc_blocker_tl {#1}
      {
        \tl_put_right:Nn \l__mfirstuc_blocker_tl {#1}
      }
      \MFUexcl { #1 }
    }
  }
```

Test if the given command is in the blocker list.

```
\cs_new:Npn \tl_if_mfirstuc_blocker:nTF #1 #2 #3
 {
   \tl_if_in:NnTF \l__mfirstuc_blocker_tl {#1} {#2} {#3}
 }
```

Is there a way to do this without `\expandafter`?

```
\cs_new:Npn \tl_if_mfirstuc_blocker:VTF #1 #2 #3
 {
   \expandafter \tl_if_mfirstuc_blocker:nTF \expandafter {#1} {#2} {#3}
 }
```

\makefirstuc

$\boxed{\texttt{\makefirstuc\{}\langle \textit{text} \rangle\texttt{\}}}$

```
\newrobustcmd{\makefirstuc}[1]{%
  \tl_set:Nn \l__mfirstuc_stack_tl {#1}
  \tl_if_empty:NF \l__mfirstuc_stack_tl
  {
    \tl_if_single:NTF \l__mfirstuc_stack_tl
    {
      \glsmakefirstuc{#1}
    }
    {
      \tl_if_head_is_group:nTF {#1}
      {
        \glsmakefirstuc{#1}
      }
      {
        \tl_if_head_eq_catcode:nNTF {#1} \relax
        {
          \tl_set:NV \l_tmpa_tl \l__mfirstuc_stack_tl
          \tl_set:Nx \l__mfirstuc_head_tl { \tl_head:V \l_tmpa_tl }
          \tl_set:Nx \l__mfirstuc_stack_tl { \tl_tail:N \l__mfirstuc_stack_tl }
          \tl_if_eq:NnTF \l__mfirstuc_head_tl { \protect }
          {
            \tl_set:NV \l_tmpa_tl \l__mfirstuc_stack_tl
            \tl_set:Nx \l__mfirstuc_head_tl { \tl_head:V \l_tmpa_tl }
            \tl_set:Nx \l__mfirstuc_stack_tl { \tl_tail:N \l__mfirstuc_stack_tl }
          }
```

```
          {}
          \tl_if_mfirstuc_blocker:VTF \l__mfirstuc_head_tl
          { #1 }
          {
            \let \l_tmpa_tl \l__mfirstuc_stack_tl
            \tl_set:Nx \l__mfirstuc_next_tl { \tl_head:N \l_tmpa_tl }
            \tl_if_single:NTF \l__mfirstuc_next_tl
            {
              \glsmakefirstuc{#1}
            }
            {
              \tl_set:Nx \l__mfirstuc_stack_tl { \tl_tail:N \l__mfirstuc_stack_tl }
              \tl_clear:N \l_tmpa_tl
              \tl_put_right:Nn \l_tmpa_tl { \__mfirstuc_inner:nnn }
              \tl_put_right:NV \l_tmpa_tl \l__mfirstuc_head_tl
              \tl_put_right:Nx  \l_tmpa_tl
              {
                { \expandonce \l__mfirstuc_next_tl }
                { \expandonce \l__mfirstuc_stack_tl }
              }
              \tl_use:N \l_tmpa_tl
            }
          }
        }
      }
      {
        \glsmakefirstuc{#1}
      }
    }
  }
 }
}

\cs_new_protected:Npn \__mfirstuc_inner:nnn #1 #2 #3
 {
```

Mappings trump exclusions.

```
    \mfirstuc_get_map:NN #1 \l_tmpb_tl
    \quark_if_no_value:NTF \l_tmpb_tl
    {
      \mfirstuc_if_excluded:NTF #1
      { #1{#2}\makefirstuc{#3} }
      { #1{\makefirstuc{#2}}#3 }
    }
    {
      \l_tmpb_tl {#2} #3
    }
  }
```

Mappings.

```
 \prop_new:N \l__mfirstuc_mappings_prop
```

\MFUaddmap Adds a mapping from a control sequence to the appropriate sentence case version. No map is added if either argument is empty.

```
\newrobustcmd*{\MFUaddmap}[2]{%
  \tl_if_empty:nF { #1 }
  {
    \tl_if_empty:nF { #2 }
    {
      \prop_put:Nnn \l__mfirstuc_mappings_prop {#1} {#2}
    }
  }
  \MFUexcl { #1 }
  \MFUblocker { #2 }
}

\cs_new:Npn \mfirstuc_get_map:NN #1 #2
 {
    \prop_get:NnN \l__mfirstuc_mappings_prop {#1} #2
 }
```

\MFUsaveatend Save exclusions, blockers and mappings in the aux file, if required by some external tool, but delays action to the end of the document.

```
\newcommand{\MFUsaveatend}{
  \AtEndDocument { \@MFUsavetoaux }
```

Disable this command.

```
  \let\MFUsaveatend\relax
}
```

\MFUsave As above but do now.

```
\newcommand{\MFUsave}{
  \@MFUsavetoaux
```

Disable further attempts.

```
  \let\@MFUsavetoaux\relax
  \let\MFUsave\relax
  \let\MFUsaveatend\relax
}
```

\@MFUsavetoaux Save exclusions, blockers and mappings in the aux file, if required by some external tool.

```
\newcommand{\@MFUsavetoaux}{
  \protected@write\@auxout{}
    { \tl_to_str:n { \providecommand{\@mfu@excls}[1]{} } }
  \protected@write\@auxout{}
    { \tl_to_str:n { \providecommand{\@mfu@blockers}[1]{} } }
  \protected@write\@auxout{}
    { \tl_to_str:n { \providecommand{\@mfu@mappings}[1]{} } }
```

Save current values.

```
  \protected@write\@auxout{}
```

```
    {
      \string\@mfu@excls { \tl_to_str:V \l_text_case_exclude_arg_tl }
    }
    \protected@write\@auxout{}
    {
      \string\@mfu@blockers { \tl_to_str:V \l__mfirstuc_blocker_tl }
    }
    \protected@write\@auxout{}
    {
      \string\@mfu@mappings
      { \expandonce { \prop_to_keyval:N \l__mfirstuc_mappings_prop } }
    }
  }

  \ExplSyntaxOff
```

The remaining code hasn't been converted to LaTeX3 yet, but may in the future, so don't rely on internal commands.

\emakefirstuc Fully expand argument before applying case-change.

```
\newrobustcmd*{\emakefirstuc}[1]{%
  \protected@edef\@MFU@caparg{#1}%
  \expandafter\makefirstuc\expandafter{\@MFU@caparg}%
}
```

\xmakefirstuc

```
\newcommand*{\xmakefirstuc}[1]{\expandafter\makefirstuc\expandafter{#1}}
```

\MFUskippunc

```
\newrobustcmd*{\MFUskippunc}[1]{#1}
\MFUexcl{\MFUskippunc}
```

Title case commands. Each word has the first letter converted to upper case using \MFUcapwordfirstuc. This uses \makefirstuc by default. It may be redefined to use \MFUsentencecase or \text_titlecase:n as applicable.

\MFUcapwordfirstuc

```
\newcommand*{\MFUcapwordfirstuc}[1]{%
  \makefirstuc{#1}%
}
```

\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).

```
\newrobustcmd*{\capitalisewords}[1]{%
  \let\@mfu@checkword\@gobble
  \def\gls@add@space{}%
  \let\@mfu@domakefirstuc\@MFU@dofirst@capword
  \mfu@capitalisewords#1 \@nil\mfu@endcap
}
```

7

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

```
\newcommand*{\@mfu@mid@capitalisewords}[1]{%
  \def\gls@add@space{}%
  \let\@mfu@domakefirstuc\@MFU@do@capword
  \mfu@capitalisewords#1 \@nil\mfu@endcap
}

\def\mfu@capitalisewords#1 #2\mfu@endcap{%
  \def\mfu@cap@first{#1}%
  \def\mfu@cap@second{#2}%
  \gls@add@space
  \@mfu@checkword{#1}%
  \@mfu@domakefirstuc{#1}%
  \def\gls@add@space{ }%
  \ifx\mfu@cap@second\@nnil
    \let\next@mfu@cap\mfu@noop
  \else
    \let\next@mfu@cap\mfu@capitalisewords
    \let\@mfu@checkword\mfu@checkword
  \fi
  \next@mfu@cap#2\mfu@endcap
}
\def\mfu@noop#1\mfu@endcap{}
```

`\ifMFUhyphen` Split on hyphens.

```
\newif\ifMFUhyphen
\MFUhyphenfalse
```

`\@MFU@do@capword`

```
\newcommand*{\@MFU@do@capword}[1]{%
  \let\@MFU@cappingword\MFUcapword
  \@MFUwordbreak#1\MFUwordbreak{}\@nil\@endMFUwordbreak
}
```

`\@MFU@dofirst@capword`

```
\newcommand*{\@MFU@dofirst@capword}[1]{%
  \let\@MFU@cappingword\MFUcapword
  \@MFUfirstwordbreak#1\MFUwordbreak{}\@nil\@endMFUfirstwordbreak
}
```

`\MFUcapword`

```
\newcommand*{\MFUcapword}[1]{%
  \ifMFUhyphen
   \MFUhyphencapword{#1}%
  \else
   \MFUcapwordfirstuc{#1}%
  \fi
}
```

8

\MFUwordbreak Markup a character (or command) that should be treated as a word break.

```
\newrobustcmd*{\MFUwordbreak}[1]{#1}
```

\MFUhyphencapword

```
\newcommand*{\MFUhyphencapword}[1]{%
  \@MFUcapword#1-\@nil\@endMFUcapword
}
```

\@MFUcapword

```
\def\@MFUcapword#1-#2\@endMFUcapword{%
  \MFUcapwordfirstuc{#1}%
  \def\@mfu@argii{#2}%
  \ifx\@mfu@argii\@nnil
  \else
    -\@MFUcapword#2\@endMFUcapword
  \fi
}
```

\@MFUwordbreak

```
\def\@MFUwordbreak#1\MFUwordbreak#2#3\@endMFUwordbreak{%
  \def\@mfu@argiii{#3}%
  \ifx\@mfu@argiii\@nnil
    \@MFU@cappingword{#1}%
  \else
    \let\@MFU@cappingword\@MFU@wordbreakcapword
    \@MFU@cappingword{#1}%
    #2\MFUwordbreak#3\@endMFUwordbreak
  \fi
}
```

\@MFU@wordbreakcapword

```
\newcommand*{\@MFU@wordbreakcapword}[1]{%
  \mfu@checkword{#1}%
  \@mfu@domakefirstuc{#1}%
}
```

\@MFUfirstwordbreak

```
\def\@MFUfirstwordbreak#1\MFUwordbreak#2#3\@endMFUfirstwordbreak{%
  \def\@mfu@argiii{#3}%
  \ifx\@mfu@argiii\@nnil
    \@MFU@cappingword{#1}%
  \else
    \let\@MFU@cappingword\@MFU@firstwordbreakcapword
    \@MFU@cappingword{#1}%
    #2\@MFUfirstwordbreak#3\@endMFUfirstwordbreak
  \fi
}
```

`\@MFU@firstwordbreakcapword`

```
\newcommand*{\@MFU@firstwordbreakcapword}[1]{%
  \@mfu@domakefirstuc{#1}%
}
```

`\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.

```
\newcommand*\mfu@checkword[1]{%
  \def\mfu@checkword@arg{#1}%
  \let\@mfu@domakefirstuc\@MFU@do@capword
  \forlistloop\mfu@checkword@do\@mfu@nocaplist
}
```

`\mfu@checkword@do` Handler for loop used by `\mfu@checkword`

```
\newcommand*{\mfu@checkword@do}[1]{%
  \def\mfu@checkword@argii{#1}%
  \ifdefequal\mfu@checkword@arg\mfu@checkword@argii
  {%
    \let\@mfu@domakefirstuc\@firstofone
    \listbreak
  }%
  {%
```

Need to check if `\@mfu@terminator` has been tacked onto the end.

```
    \def\mfu@checkword@argii{#1{\@mfu@terminator}}%
    \ifdefequal\mfu@checkword@arg\mfu@checkword@argii
    {%
      \let\@mfu@domakefirstuc\@firstofone
      \listbreak
    }%
    {}%
  }%
}
```

`\@mfu@nocaplist` List of words that shouldn't be capitalised.

```
\newcommand*{\@mfu@nocaplist}{}
```

`\MFUnocap` Provide the user with a means to add a word to the list. (No expansion performed on the argument.)

```
\newcommand*{\MFUnocap}[1]{\listadd{\@mfu@nocaplist}{#1}}
```

`\gMFUnocap` Global version.

```
\newcommand*{\gMFUnocap}[1]{\listgadd{\@mfu@nocaplist}{#1}}
```

`\MFUclear` Clear the list

```
\newcommand*{\MFUclear}{\renewcommand*{\@mfu@nocaplist}{}}
```

10

\xcapitalisewords   Short-cut command:

```
\newcommand*{\xcapitalisewords}[1]{%
  \expandafter\capitalisewords\expandafter{#1}%
}
```

\ecapitalisewords   Fully expand argument before applying `\capitalisewords`

```
\newrobustcmd*{\ecapitalisewords}[1]{%
  \protected@edef\@MFU@caparg{#1}%
  \expandafter\capitalisewords\expandafter{\@MFU@caparg}%
}
```

\@mfu@nil

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

\@mfu@nnil

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

\capitalisefmtwords   Like `\capitalisewords` but assumes that the phrase contains formatting text-block commands.

```
\newcommand*{\capitalisefmtwords}{%
  \@ifstar\@s@capitalisefmtwords\@ns@capitalisefmtwords
}
```

\@s@capitalisefmtwords   Starred form.

```
\newcommand*{\@s@capitalisefmtwords}[1]{%
  \@@s@checkgrp@@capitalisefmtwords#1\@nil
}
```

\@@s@checkgrp@capitalisefmtwords

```
\newcommand*{\@@s@checkgrp@@capitalisefmtwords}{%
  \@ifnextchar\bgroup{\@@s@nocs@capitalisefmtwords}{\@@s@capitalisefmtwords}%
}
```

\@@s@nocs@capitalisefmtwords   Argument starts with a group.

```
\def\@@s@nocs@capitalisefmtwords#1\@nil{%
  \capitalisewords{#1}%
}
```

\@@s@capitalisefmtwords

```
\newcommand*{\@@s@capitalisefmtwords}[1]{%
  \def\gls@argi{#1}%
  \ifx\gls@argi\@MFU@protect
```

Skip `\protect`

```
    \let\@mfu@capfmtwordsnext\@@s@capitalisefmtwords
  \else
    \ifx\gls@argi\@nnil
```

Nothing to do.
```
              \let\@mfu@capfmtwordsnext\relax
          \else
            \def\@gls@tmp{\ #1}%
            \@onelevel@sanitize\@gls@tmp
            \expandafter\@gls@checkcs\@gls@tmp\relax\relax
            \if@glscs
               \def\@mfu@capfmtwordsnext{\@@s@cs@capitalisefmtwords#1}%
            \else
               \def\@mfu@capfmtwordsnext{\@@s@nocs@capitalisefmtwords#1}%
            \fi
          \fi
      \fi
      \@mfu@capfmtwordsnext
  }
```

\@@s@cs@capitalisefmtwords
```
\newcommand*{\@@s@cs@capitalisefmtwords}[1]{%
  \@ifnextchar\bgroup
   {\@@s@cs@g@capitalisefmtwords#1}%
   {\@@s@cs@ng@capitalisefmtwords#1}%
}
```

\@@s@cs@g@capitalisefmtwords
```
\def\@@s@cs@g@capitalisefmtwords#1#2#3\@nil{%
  #1{\capitalisewords{#2}}%
  \ifblank{#3}%
  {}%
  {%
    \def\gls@add@space{}%
    \let\@mfu@checkword\mfu@checkword
    \mfu@capitalisewords#3 \@nil\mfu@endcap
  }%
}
```

\@@s@cs@ng@capitalisefmtwords
```
\def\@@s@cs@ng@capitalisefmtwords#1#2\@nil{%
  \capitalisewords{#1#2}%
}
```

\@ns@capitalisefmtwords Unstarred form.
```
\newcommand*{\@ns@capitalisefmtwords}[1]{%
  \let\@mfu@checkword\@gobble
  \def\gls@add@space{}%
  \@capitalisefmtwords#1 \@mfu@nil{} \unskip
}
```

\@capitalisefmtwords Take care of group at the start of the word.
```
\newcommand*{\@capitalisefmtwords}{%
```

12

```
      \@ifnextchar\bgroup{\@capitalise@g@fmtwords}{\@capitalise@ng@fmtwords}%
    }
    \newcommand*{\@capitalise@g@fmtwords}[1]{%
      \@capitalise@ng@fmtwords{{#1}}%
    }
    \def\@mfu@ifnil#1#2\@mfu@end@nil#3#4{%
      \def\gls@argi{#1}%
      \ifx\gls@argi\@mfu@nnil
        #3%
      \else
        #4%
      \fi
    }
    \def\@capitalise@ng@fmtwords#1 {%
      \@mfu@ifnil#1\relax\relax\@mfu@end@nil
      {%
        \def\@mfu@capfmtwordsnext{#1}%
      }%
      {%
        \def\@mfu@capfmtwordsnext{%
          \@capitalise@fmtwords#1\@mfu@nil
          \def\gls@add@space{ }%
          \let\@mfu@checkword\mfu@checkword
          \@@cap@nofmtphrase}%
      }%
      \@mfu@capfmtwordsnext
    }%
```

\@capitalise@fmtwords

```
    \def\@capitalise@fmtwords#1{%
      \@mfu@ifnil#1\relax\relax\@mfu@end@nil
      {%
```

Do nothing.

```
        \def\@mfu@capfmtwordsnext{#1}%
      }%
      {%
        \def\gls@argi{#1}%
        \ifx\gls@argi\@MFU@protect
```

Skip \protect

```
          \def\@mfu@capfmtwordsnext{\@capitalise@fmtwords}%
        \else
          \def\@gls@tmp{\ #1}%
          \@onelevel@sanitize\@gls@tmp
          \expandafter\@gls@checkcs\@gls@tmp\relax\relax
          \if@glscs
            \def\@mfu@capfmtwordsnext{\@@capitalisefmtwords#1}%
          \else
            \ifx\gls@argi\@empty
```

```
                    \def\@mfu@capfmtwordsnext{\@@capnofmtphrase{}}%
                  \else
                    \def\@mfu@capfmtwordsnext{\@@capnofmtphrase#1}%
                  \fi
                \fi
              \fi
            }%
            \@mfu@capfmtwordsnext
          }
```

\@@capitalisefmtwords   Check for a group following argument.

```
\def\@@capitalisefmtwords#1{%
  \@ifnextchar\bgroup{\@@capfmtsubphrase#1}{\@@capnofmtphrase#1}%
}
```

\@@capnofmtphrase

```
\def\@@capnofmtphrase{%
  \@ifnextchar\bgroup{\@@cap@g@nofmtphrase}{\@@cap@ng@nofmtphrase}%
}
```

\@@cap@ng@nofmtphrase

```
\def\@@cap@ng@nofmtphrase#1\@mfu@nil{%
  \gls@add@space
  \@mfu@mid@capitalisewords{#1}%
  \def\gls@add@space{ }%
}
```

\@@cap@g@nofmtphrase

```
\def\@@cap@g@nofmtphrase#1{%
  \@@cap@ng@nofmtphrase{{#1}}%
}
```

\@mfu@terminator

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

\@@capfmtsubphrase

```
\def\@@capfmtsubphrase#1#2#3{%
```

Protect any trailing space at the end of the second argument.

```
  #1{\@capitalisefmtwords#2{\@mfu@terminator} \@mfu@nil{} \unskip}%
  \def\gls@add@space{ }%
  \let\@mfu@checkword\mfu@checkword
  \@mfu@ifnil#1\relax\relax\@mfu@end@nil
  {\let\@mfu@fmtsubphrasenext\relax}%
  {\def\@mfu@fmtsubphrasenext{#3}}%
  \@mfu@fmtsubphrasenext
}
```

14

\@@cap@nofmtphrase

```
\def\@@cap@nofmtphrase#1 {%
  \@mfu@ifnil#1\relax\relax\@mfu@end@nil
  {%
    \def\@mfu@capnofmtwordsnext{#1 }%
  }%
  {%
    \def\@mfu@capnofmtwordsnext{%
      \@capitalisefmtwords#1 \@mfu@nil{} \unskip
      \def\gls@add@space{ }%
      \let\@mfu@checkword\mfu@checkword
      \@capitalisefmtwords
    }%
  }%
  \@mfu@capnofmtwordsnext
}
```

\xcapitalisefmtwords   Short-cut command:

```
\newcommand*{\xcapitalisefmtwords}{%
  \@ifstar\@s@xcapitalisefmtwords\@ns@xcapitalisefmtwords
}
```

\@s@xcapitalisefmtwords   Starred version

```
\newcommand*{\@s@xcapitalisefmtwords}[1]{%
  \expandafter\@s@capitalisefmtwords\expandafter{#1}%
}
```

\@ns@xcapitalisefmtwords   Unstarred version

```
\newcommand*{\@ns@xcapitalisefmtwords}[1]{%
  \expandafter\@ns@capitalisefmtwords\expandafter{#1}%
}
```

\ecapitalisefmtwords

```
\newrobustcmd*{\ecapitalisefmtwords}{%
  \@ifstar\@s@ecapitalisefmtwords\@ns@ecapitalisefmtwords
}
```

\@s@ecapitalisefmtwords   Starred version

```
\newcommand*{\@s@ecapitalisefmtwords}[1]{%
  \protected@edef\@MFU@caparg{#1}%
  \expandafter\@s@capitalisefmtwords\expandafter{\@MFU@caparg}%
}
```

\@ns@ecapitalisefmtwords   Unstarred version

```
\newcommand*{\@ns@ecapitalisefmtwords}[1]{%
  \protected@edef\@MFU@caparg{#1}%
  \expandafter\@ns@capitalisefmtwords\expandafter{\@MFU@caparg}%
}
```

The remaining code is redundant and will be removed in future.

\@org@makefirstuc This is the original definition of \makefirstuc. It will be removed in later versions. Syntax:

> \makefirstuc{⟨*text*⟩}

```
\newif\if@glscs
\newtoks\@glsmfirst
\newtoks\@glsmrest
\newrobustcmd*{\@org@makefirstuc}[1]{%
```
Check if argument starts with \protect which needs to be skipped.
```
  \@MFU@p@makefirstuc#1\relax\relax\relax\@end@MFU@protected{#1}%
}
```

\@MFU@protect

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

\@MFU@skippunc

```
\newcommand*{\@MFU@skippunc}{\MFUskippunc}
```

\@MFU@p@makefirstuc Internal command to do the required action.
```
\def\@MFU@p@makefirstuc#1#2#3\@end@MFU@protected#4{%
```
Test if the first argument is \protect.
```
  \def\gls@argi{#1}%
  \ifx\gls@argi\@MFU@protect
   \@makefirstuc{#2#3}%
  \else
```
v2.07+ also check for \MFUskippunc
```
    \ifx\gls@argi\@MFU@skippunc
      #2\@makefirstuc{#3}%
    \else
      \@makefirstuc{#4}%
    \fi
  \fi
 }
```

\@makefirstuc Internal command to do the required action.
```
\newcommand*{\@makefirstuc}[1]{%
  \def\gls@argi{#1}%
  \ifx\gls@argi\@empty
```
If the argument is empty, do nothing.
```
  \else
```

16

```
        \def\@gls@tmp{\ #1}%
        \@onelevel@sanitize\@gls@tmp
        \expandafter\@gls@checkcs\@gls@tmp\relax\relax
        \if@glscs
```

Check if we have two sequential control sequences.

```
          \expandafter\def\expandafter\@gls@tmp\expandafter{\expandafter
            \ \@gobble #1}%
          \@onelevel@sanitize\@gls@tmp
          \expandafter\@gls@checkcs\@gls@tmp\relax\relax
          \if@glscs\@glscsfalse\else\@glscstrue\fi
        \fi
        \if@glscs
          \@gls@getbody #1{}\@nil
          \ifx\@gls@rest\@empty
            \glsmakefirstuc{#1}%
          \else
            \expandafter\@gls@split\@gls@rest\@nil
            \ifx\@gls@first\@empty
              \glsmakefirstuc{#1}%
            \else
```

Does `\@gls@first` start with a control sequence?

```
              \edef\@gls@tmp\expandafter{\noexpand\ \expandonce\@gls@first}%
              \@onelevel@sanitize\@gls@tmp
              \expandafter\@gls@checkcs\@gls@tmp\relax\relax
              \if@glscs
                \edef\@gls@domfirstuc{\expandonce\@gls@body
                 {\noexpand\makefirstuc{\expandonce\@gls@first
                   \expandonce\@gls@rest}%
                 }%
                }%
              \else
                \expandafter\@glsmfirst\expandafter{\@gls@first}%
                \expandafter\@glsmrest\expandafter{\@gls@rest}%

                \edef\@gls@domfirstuc{\noexpand\@gls@body
                  {\noexpand\glsmakefirstuc{\the\@glsmfirst}}%
                  \the\@glsmrest}%
              \fi
              \@gls@domfirstuc
          \fi
        \fi
        \else
          \glsmakefirstuc{#1}%
        \fi
      \fi
    }
```

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

```
\ifdef\dtl@getfirst@UTFviii
{%
  \def\@gls@split#1#2\@nil{%
   \ifbool{@dtl@utf8}
     {%
       \expandafter\dtl@if@two@octets#1#2\relax\dtl@end@if@two@octets
       {%
         \dtl@getfirst@UTFviii#1#2\@nil\end@dtl@getfirst@UTFviii
         \let\@gls@first\dtl@first
         \let\@gls@rest\dtl@rest
       }%
       {%
         \def\@gls@first{#1}\def\@gls@rest{#2}%
       }%
     }%
     {%

       \def\@gls@first{#1}\def\@gls@rest{#2}%
     }%
   }
}%
{%
  \def\@gls@split#1#2\@nil{%
    \def\@gls@first{#1}\def\@gls@rest{#2}%
  }
}

\def\@gls@checkcs#1 #2#3\relax{%
  \def\@gls@argi{#1}\def\@gls@argii{#2}%
  \ifx\@gls@argi\@gls@argii
    \@glscstrue
  \else
    \@glscsfalse
  \fi
}
```

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

\@mfu@applytofirst

```
\ifdef\dtl@getfirst@UTFviii
{%
  \def\@mfu@applytofirst#1#2\@mfu@end@applytofirst#3{%
    \expandafter\dtl@if@two@octets#1#2\relax\dtl@end@if@two@octets
    {%
      \dtl@getfirst@UTFviii#1#2\@nil\end@dtl@getfirst@UTFviii
      \expandafter#3\expandafter{\dtl@first}\dtl@rest
    }%
    {%
```

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

```
            #3{#1}#2%
          }%
        }%
      }%
      {%
```
First argument needs to be grouped in case it consists of more than one token.
```
        \def\@mfu@applytofirst#1#2\@mfu@end@applytofirst#3{%
            #3{#1}#2%
        }
      }
```

> $\boxed{\texttt{\textbackslash MFUapplytofirst\{}\langle cs \rangle\texttt{\}\{}\langle text \rangle\texttt{\}}}$

Apply $\langle cs \rangle$ to first token of text, or first UTF-8 character if datatool-base v2.24 above has been loaded with the utf8 support enabled. Deprecated as from v2.08.
```
    \ifdef\dtl@getfirst@UTFviii
    {%
      \newcommand*{\MFUapplytofirst}[2]{%
        \ifbool{@dtl@utf8}
        {%
          \ifblank{#2}%
          {#1{#2}}%
          {%
            \ifblank{#1}%
            {#1#2}%
            {%
              \@mfu@applytofirst#2\@mfu@end@applytofirst#1\relax
            }%
          }%
        }%
        {%
          #1#2%
        }%
      }
    }%
    {%
      \newcommand*{\MFUapplytofirst}[2]{%
          #1#2%
      }
    }
```

Deprecated.
```
    \def\@gls@makefirstuc#1{\MFUsentencecase{#1}}
```

Get the first grouped argument and store in \@gls@body.
```
    \def\@gls@getbody#1#{\def\@gls@body{#1}\@gls@gobbletonil}
```
Scoup up everything to \@nil and store in \@gls@rest:
```
    \def\@gls@gobbletonil#1\@nil{\def\@gls@rest{#1}}
```

3.1 Mfirstuc-english Documented Code

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{mfirstuc-english}[2022/10/14 v2.08 (NLCT)]
```

Load mfirstuc if not already loaded:
```
\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.
```
\MFUnocap{a}
\MFUnocap{an}
\MFUnocap{and}
\MFUnocap{but}
\MFUnocap{for}
\MFUnocap{in}
\MFUnocap{of}
\MFUnocap{or}
\MFUnocap{no}
\MFUnocap{nor}
\MFUnocap{so}
\MFUnocap{some}
\MFUnocap{the}
\MFUnocap{with}
\MFUnocap{yet}
```

4 Rollback v2.07 (mfirstuc-2021-10-15.sty)

Version 2.07 preserved for rollback.
```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{mfirstuc}[2021/10/15 v2.07 (NLCT)]
\RequirePackage{etoolbox}
\newif\if@glscs
\newtoks\@glsmfirst
\newtoks\@glsmrest
\newrobustcmd*{\makefirstuc}[1]{%
  \@MFU@p@makefirstuc#1\relax\relax\relax\@end@MFU@protected{#1}%
}
\newcommand*{\@MFU@protect}{\protect}
\newcommand*{\@MFU@skippunc}{\MFUskippunc}
\newrobustcmd*{\MFUskippunc}[1]{#1}
\def\@MFU@p@makefirstuc#1#2#3\@end@MFU@protected#4{%
  \def\gls@argi{#1}%
  \ifx\gls@argi\@MFU@protect
   \@makefirstuc{#2#3}%
  \else
    \ifx\gls@argi\@MFU@skippunc
      #2\@makefirstuc{#3}%
    \else
      \@makefirstuc{#4}%
    \fi
```

```
    \fi
}
\newcommand*{\@makefirstuc}[1]{%
  \def\gls@argi{#1}%
  \ifx\gls@argi\@empty
  \else
    \def\@gls@tmp{\ #1}%
    \@onelevel@sanitize\@gls@tmp
    \expandafter\@gls@checkcs\@gls@tmp\relax\relax
    \if@glscs
     \expandafter\def\expandafter\@gls@tmp\expandafter{\expandafter
       \ \@gobble #1}%
     \@onelevel@sanitize\@gls@tmp
     \expandafter\@gls@checkcs\@gls@tmp\relax\relax
     \if@glscs\@glscsfalse\else\@glscstrue\fi
    \fi
    \if@glscs
      \@gls@getbody #1{}\@nil
      \ifx\@gls@rest\@empty
        \glsmakefirstuc{#1}%
      \else
        \expandafter\@gls@split\@gls@rest\@nil
        \ifx\@gls@first\@empty
          \glsmakefirstuc{#1}%
        \else
          \edef\@gls@tmp\expandafter{\noexpand\ \expandonce\@gls@first}%
          \@onelevel@sanitize\@gls@tmp
          \expandafter\@gls@checkcs\@gls@tmp\relax\relax
          \if@glscs
            \edef\@gls@domfirstuc{\expandonce\@gls@body
             {\noexpand\makefirstuc{\expandonce\@gls@first
              \expandonce\@gls@rest}%
             }%
            }%
          \else
            \expandafter\@glsmfirst\expandafter{\@gls@first}%
            \expandafter\@glsmrest\expandafter{\@gls@rest}%
            \edef\@gls@domfirstuc{\noexpand\@gls@body
              {\noexpand\glsmakefirstuc{\the\@glsmfirst}}%
              \the\@glsmrest}%
          \fi
          \@gls@domfirstuc
        \fi
      \fi
    \else
      \glsmakefirstuc{#1}%
    \fi
  \fi
}
\ifdef\dtl@getfirst@UTFviii
```

```
{%
  \def\@gls@split#1#2\@nil{%
   \ifbool{@dtl@utf8}
     {%
       \expandafter\dtl@if@two@octets#1#2\relax\dtl@end@if@two@octets
       {%
         \dtl@getfirst@UTFviii#1#2\@nil\end@dtl@getfirst@UTFviii
         \let\@gls@first\dtl@first
         \let\@gls@rest\dtl@rest
       }%
       {%
         \def\@gls@first{#1}\def\@gls@rest{#2}%
       }%
     }%
     {%
       \def\@gls@first{#1}\def\@gls@rest{#2}%
     }%
   }
}%
{%
  \def\@gls@split#1#2\@nil{%
    \def\@gls@first{#1}\def\@gls@rest{#2}%
  }
}
\def\@gls@checkcs#1 #2#3\relax{%
  \def\@gls@argi{#1}\def\@gls@argii{#2}%
  \ifx\@gls@argi\@gls@argii
    \@glscstrue
  \else
    \@glscsfalse
  \fi
}
\ifdef\dtl@getfirst@UTFviii
{%
  \def\@mfu@applytofirst#1#2\@mfu@end@applytofirst#3{%
    \expandafter\dtl@if@two@octets#1#2\relax\dtl@end@if@two@octets
    {%
      \dtl@getfirst@UTFviii#1#2\@nil\end@dtl@getfirst@UTFviii
      \expandafter#3\expandafter{\dtl@first}\dtl@rest
    }%
    {%
      #3{#1}#2%
    }%
  }%
}%
{%
  \def\@mfu@applytofirst#1#2\@mfu@end@applytofirst#3{#3{#1}#2}
}
\ifdef\dtl@getfirst@UTFviii
{%
```

```
\newcommand*{\MFUapplytofirst}[2]{%
  \ifbool{@dtl@utf8}
  {%
    \ifblank{#2}%
    {#1{#2}}%
    {%
      \ifblank{#1}%
      {#1#2}%
      {%
        \@mfu@applytofirst#2\@mfu@end@applytofirst#1\relax
      }%
    }%
  }%
  {%
    #1#2%
  }%
}
}%
{%
  \newcommand*{\MFUapplytofirst}[2]{#1#2}
}
\def\@gls@makefirstuc#1{\MFUapplytofirst\mfirstucMakeUppercase{#1}}
\newcommand*{\mfirstucMakeUppercase}{\MakeUppercase}
\newcommand*{\glsmakefirstuc}[1]{\@gls@makefirstuc{#1}}
\def\@gls@getbody#1#{\def\@gls@body{#1}\@gls@gobbletonil}
\def\@gls@gobbletonil#1\@nil{\def\@gls@rest{#1}}
\newcommand*{\xmakefirstuc}[1]{%
  \expandafter\makefirstuc\expandafter{#1}%
}
\newrobustcmd*{\emakefirstuc}[1]{%
  \protected@edef\@MFU@caparg{#1}%
  \expandafter\makefirstuc\expandafter{\@MFU@caparg}%
}
\newrobustcmd*{\capitalisewords}[1]{%
  \let\@mfu@checkword\@gobble
  \def\gls@add@space{}%
  \let\@mfu@domakefirstuc\@MFU@dofirst@capword
  \mfu@capitalisewords#1 \@nil\mfu@endcap
}
\newcommand*{\@mfu@mid@capitalisewords}[1]{%
  \def\gls@add@space{}%
  \let\@mfu@domakefirstuc\@MFU@do@capword
  \mfu@capitalisewords#1 \@nil\mfu@endcap
}
\def\mfu@capitalisewords#1 #2\mfu@endcap{%
  \def\mfu@cap@first{#1}%
  \def\mfu@cap@second{#2}%
  \gls@add@space
  \@mfu@checkword{#1}%
  \@mfu@domakefirstuc{#1}%
```

```
    \def\gls@add@space{ }%
    \ifx\mfu@cap@second\@nnil
      \let\next@mfu@cap\mfu@noop
    \else
      \let\next@mfu@cap\mfu@capitalisewords
      \let\@mfu@checkword\mfu@checkword
    \fi
    \next@mfu@cap#2\mfu@endcap
}
\def\mfu@noop#1\mfu@endcap{}
\newif\ifMFUhyphen
\MFUhyphenfalse
\newcommand*{\@MFU@do@capword}[1]{%
  \let\@MFU@cappingword\MFUcapword
  \@MFUwordbreak#1\MFUwordbreak{}\@nil\@endMFUwordbreak
}
\newcommand*{\@MFU@dofirst@capword}[1]{%
  \let\@MFU@cappingword\MFUcapword
  \@MFUfirstwordbreak#1\MFUwordbreak{}\@nil\@endMFUfirstwordbreak
}
\newcommand*{\MFUcapword}[1]{%
  \ifMFUhyphen
   \MFUhyphencapword{#1}%
  \else
   \MFUcapwordfirstuc{#1}%
  \fi
}
\newcommand*{\MFUcapwordfirstuc}[1]{%
  \makefirstuc{#1}%
}
\newrobustcmd*{\MFUwordbreak}[1]{#1}
\newcommand*{\MFUhyphencapword}[1]{%
  \@MFUcapword#1-\@nil\@endMFUcapword
}
\def\@MFUcapword#1-#2\@endMFUcapword{%
  \MFUcapwordfirstuc{#1}%
  \def\@mfu@argii{#2}%
  \ifx\@mfu@argii\@nnil
  \else
    -\@MFUcapword#2\@endMFUcapword
  \fi
}
\def\@MFUwordbreak#1\MFUwordbreak#2#3\@endMFUwordbreak{%
  \def\@mfu@argiii{#3}%
  \ifx\@mfu@argiii\@nnil
    \@MFU@cappingword{#1}%
  \else
    \let\@MFU@cappingword\@MFU@wordbreakcapword
    \@MFU@cappingword{#1}%
    #2\@MFUwordbreak#3\@endMFUwordbreak
```

```
  \fi
}
\newcommand*{\@MFU@wordbreakcapword}[1]{%
  \mfu@checkword{#1}%
  \@mfu@domakefirstuc{#1}%
}
\def\@MFUfirstwordbreak#1\MFUwordbreak#2#3\@endMFUfirstwordbreak{%
  \def\@mfu@argiii{#3}%
  \ifx\@mfu@argiii\@nnil
    \@MFU@cappingword{#1}%
  \else
    \let\@MFU@cappingword\@MFU@firstwordbreakcapword
    \@MFU@cappingword{#1}%
    #2\@MFUfirstwordbreak#3\@endMFUfirstwordbreak
  \fi
}
\newcommand*{\@MFU@firstwordbreakcapword}[1]{%
  \@mfu@domakefirstuc{#1}%
}
\newcommand*\mfu@checkword[1]{%
  \def\mfu@checkword@arg{#1}%
  \let\@mfu@domakefirstuc\@MFU@do@capword
  \forlistloop\mfu@checkword@do\@mfu@nocaplist
}
\newcommand*{\mfu@checkword@do}[1]{%
  \def\mfu@checkword@argii{#1}%
  \ifdefequal\mfu@checkword@arg\mfu@checkword@argii
  {%
    \let\@mfu@domakefirstuc\@firstofone
    \listbreak
  }%
  {%
    \def\mfu@checkword@argii{#1{\@mfu@terminator}}%
    \ifdefequal\mfu@checkword@arg\mfu@checkword@argii
    {%
      \let\@mfu@domakefirstuc\@firstofone
      \listbreak
    }%
    {}%
  }%
}
\newcommand*{\@mfu@nocaplist}{}
\newcommand*{\MFUnocap}[1]{\listadd{\@mfu@nocaplist}{#1}}
\newcommand*{\gMFUnocap}[1]{\listgadd{\@mfu@nocaplist}{#1}}
\newcommand*{\MFUclear}{\renewcommand*{\@mfu@nocaplist}{}}
\newcommand*{\xcapitalisewords}[1]{%
  \expandafter\capitalisewords\expandafter{#1}%
}
\newrobustcmd*{\ecapitalisewords}[1]{%
  \protected@edef\@MFU@caparg{#1}%
```

```
    \expandafter\capitalisewords\expandafter{\@MFU@caparg}%
}
\newcommand\@mfu@nil{\unskip}
\newcommand*\@mfu@nnil{\@mfu@nil}
\newcommand*{\capitalisefmtwords}{%
  \@ifstar\@s@capitalisefmtwords\@ns@capitalisefmtwords
}
\newcommand*{\@s@capitalisefmtwords}[1]{%
  \@@s@checkgrp@@capitalisefmtwords#1\@nil
}
\newcommand*{\@@s@checkgrp@@capitalisefmtwords}{%
  \@ifnextchar\bgroup{\@@s@nocs@capitalisefmtwords}{\@@s@capitalisefmtwords}%
}
\def\@@s@nocs@capitalisefmtwords#1\@nil{%
  \capitalisewords{#1}%
}
\newcommand*{\@@s@capitalisefmtwords}[1]{%
  \def\gls@argi{#1}%
  \ifx\gls@argi\@MFU@protect
    \let\@mfu@capfmtwordsnext\@@s@capitalisefmtwords
  \else
    \ifx\gls@argi\@nnil
      \let\@mfu@capfmtwordsnext\relax
    \else
      \def\@gls@tmp{\ #1}%
      \@onelevel@sanitize\@gls@tmp
      \expandafter\@gls@checkcs\@gls@tmp\relax\relax
      \if@glscs
        \def\@mfu@capfmtwordsnext{\@@s@cs@capitalisefmtwords#1}%
      \else
        \def\@mfu@capfmtwordsnext{\@@s@nocs@capitalisefmtwords#1}%
      \fi
    \fi
  \fi
  \@mfu@capfmtwordsnext
}
\newcommand*{\@@s@cs@capitalisefmtwords}[1]{%
  \@ifnextchar\bgroup
   {\@@s@cs@g@capitalisefmtwords#1}%
   {\@@s@cs@ng@capitalisefmtwords#1}%
}
\def\@@s@cs@g@capitalisefmtwords#1#2#3\@nil{%
  #1{\capitalisewords{#2}}%
  \ifblank{#3}%
  {}%
  {%
    \def\gls@add@space{}%
    \let\@mfu@checkword\mfu@checkword
    \mfu@capitalisewords#3 \@nil\mfu@endcap
  }%
```

```
}
\def\@@s@cs@ng@capitalisefmtwords#1#2\@nil{%
  \capitalisewords{#1#2}%
}
\newcommand*{\@ns@capitalisefmtwords}[1]{%
  \let\@mfu@checkword\@gobble
  \def\gls@add@space{}%
  \@capitalisefmtwords#1 \@mfu@nil{} \unskip
}
\newcommand*{\@capitalisefmtwords}{%
  \@ifnextchar\bgroup{\@capitalise@g@fmtwords}{\@capitalise@ng@fmtwords}%
}
\newcommand*{\@capitalise@g@fmtwords}[1]{%
  \@capitalise@ng@fmtwords{{#1}}%
}
\def\@mfu@ifnil#1#2\@mfu@end@nil#3#4{%
  \def\gls@argi{#1}%
  \ifx\gls@argi\@mfu@nnil
    #3%
  \else
    #4%
  \fi
}
\def\@capitalise@ng@fmtwords#1 {%
  \@mfu@ifnil#1\relax\relax\@mfu@end@nil
  {%
    \def\@mfu@capfmtwordsnext{#1}%
  }%
  {%
    \def\@mfu@capfmtwordsnext{%
      \@capitalise@fmtwords#1\@mfu@nil
      \def\gls@add@space{ }%
      \let\@mfu@checkword\mfu@checkword
      \@@cap@nofmtphrase}%
  }%
  \@mfu@capfmtwordsnext
}%
\def\@capitalise@fmtwords#1{%
  \@mfu@ifnil#1\relax\relax\@mfu@end@nil
  {%
    \def\@mfu@capfmtwordsnext{#1}%
  }%
  {%
    \def\gls@argi{#1}%
    \ifx\gls@argi\@MFU@protect
      \def\@mfu@capfmtwordsnext{\@capitalise@fmtwords}%
    \else
      \def\@gls@tmp{\ #1}%
      \@onelevel@sanitize\@gls@tmp
      \expandafter\@gls@checkcs\@gls@tmp\relax\relax
```

```
      \if@glscs
       \def\@mfu@capfmtwordsnext{\@@capitalisefmtwords#1}%
      \else
       \ifx\gls@argi\@empty
          \def\@mfu@capfmtwordsnext{\@@capnofmtphrase{}}%
       \else
          \def\@mfu@capfmtwordsnext{\@@capnofmtphrase#1}%
       \fi
      \fi
    \fi
  }%
  \@mfu@capfmtwordsnext
}
\def\@@capitalisefmtwords#1{%
  \@ifnextchar\bgroup{\@@capfmtsubphrase#1}{\@@capnofmtphrase#1}%
}
\def\@@capnofmtphrase{%
  \@ifnextchar\bgroup{\@@cap@g@nofmtphrase}{\@@cap@ng@nofmtphrase}%
}
\def\@@cap@ng@nofmtphrase#1\@mfu@nil{%
  \gls@add@space
  \@mfu@mid@capitalisewords{#1}%
  \def\gls@add@space{ }%
}
\def\@@cap@g@nofmtphrase#1{%
  \@@cap@ng@nofmtphrase{{#1}}%
}
\newcommand*\@mfu@terminator{\hbox{}}
\def\@@capfmtsubphrase#1#2#3{%
  #1{\@capitalisefmtwords#2{\@mfu@terminator} \@mfu@nil{} \unskip}%
  \def\gls@add@space{ }%
  \let\@mfu@checkword\mfu@checkword
  \@mfu@ifnil#1\relax\relax\@mfu@end@nil
  {\let\@mfu@fmtsubphrasenext\relax}%
  {\def\@mfu@fmtsubphrasenext{#3}}%
  \@mfu@fmtsubphrasenext
}
\def\@@cap@nofmtphrase#1 {%
  \@mfu@ifnil#1\relax\relax\@mfu@end@nil
  {%
    \def\@mfu@capnofmtwordsnext{#1 }%
  }%
  {%
    \def\@mfu@capnofmtwordsnext{%
      \@capitalisefmtwords#1 \@mfu@nil{} \unskip
      \def\gls@add@space{ }%
      \let\@mfu@checkword\mfu@checkword
      \@capitalisefmtwords
    }%
  }%
```

```
  \@mfu@capnofmtwordsnext
}
\newcommand*{\xcapitalisefmtwords}{%
  \@ifstar\@s@xcapitalisefmtwords\@ns@xcapitalisefmtwords
}
\newcommand*{\@s@xcapitalisefmtwords}[1]{%
  \expandafter\@s@capitalisefmtwords\expandafter{#1}%
}
\newcommand*{\@ns@xcapitalisefmtwords}[1]{%
  \expandafter\@ns@capitalisefmtwords\expandafter{#1}%
}
\newrobustcmd*{\ecapitalisefmtwords}{%
  \@ifstar\@s@ecapitalisefmtwords\@ns@ecapitalisefmtwords
}
\newcommand*{\@s@ecapitalisefmtwords}[1]{%
  \protected@edef\@MFU@caparg{#1}%
  \expandafter\@s@capitalisefmtwords\expandafter{\@MFU@caparg}%
}
\newcommand*{\@ns@ecapitalisefmtwords}[1]{%
  \protected@edef\@MFU@caparg{#1}%
  \expandafter\@ns@capitalisefmtwords\expandafter{\@MFU@caparg}%
}
```

Change History

30

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the definition; numbers in roman refer to the pages where the entry is used.

33