Grzegorz `Natror' Murzynowski

The gmiflink Package*

Written by Grzegorz 'Natror' Murzynowski,
natror at o2 dot pl

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{gmiflink}
    [2006/08/16 v0.97 Conditionally hyperlinking package (GM)]
```

Introduction, usage

This package protects you against an error when a link is dangling and typesets some plain text instead of a hyperlink then. It is intended for use with the hyperref package. Needs *two* LaTeX runs.

I used it for typesetting the names of the objects in a documentation of a computer program. If the object had been defined a `\hyperlink` to its definition was made, otherwise a plain object's name was typeset. I also use this package in authomatic making of hyperlinking indexes.

The package provides the macros `\gmiflink`, `\gmifref` and `\gmhypertarget` for conditional making of hyperlinks in your document.

`\gmhypertarget` `\gmhypertarget[`⟨*name*⟩`]{`⟨*text*⟩`}` makes a `\hypertarget{`⟨*@name*⟩`}{`⟨*text*⟩`}` and a `\label{`⟨*@name*⟩`}`.

`\gmiflink` `\gmiflink[`⟨*name*⟩`]{`⟨*text*⟩`}` makes a `\hyperlink{`⟨*@name*⟩`}{`⟨*text*⟩`}` to a proper hypertarget if the corresponding *label* exists, otherwise it typesets ⟨*text*⟩.

`\gmifref` `\gmifref[`⟨*name*⟩`]{`⟨*text*⟩`}` makes a (hyper-) `\ref{`⟨*@name*⟩`}` to the given label if the label exists, otherwise it typesets ⟨*text*⟩.

The ⟨*@name*⟩ argument is just ⟨*name*⟩ if the ⟨*name*⟩ is given, otherwise it's ⟨*text*⟩ in all three macros.

For the example(s) of use, examine the gmiflink.sty file, lines 45–58.

Installation

Unpack the gmiflink-tds.zip (this is an archive conforming the TDS standard, see CTAN/tds/tds.pdf) in a texmf directory or put the gmiflink.sty somewhere in the texmf/tex/latex branch on your own. (Creating a texmf/tex/latex/gm directory may be advisable if you consider using other packages written by me.)

Then you should refresh your TeX distribution's files' database most probably.

[*] This file has version number v0.97 dated 2006/08/16.

Contents of the gmiflink.zip archive

The distribution of the gmiflink package consists of the following four files and a TDS-compliant archive.

gmiflink.sty
README
gmiflinkDoc.tex
gmiflinkDoc.pdf
gmiflink.tds.zip

Compiling the Documentation

The last of the above files (the .pdf, i.e., *this file*) is a documentation compiled from the .sty file by running LaTeX on the gmiflinkDoc.tex file. Compiling the documentation requires the packages: gmdoc (gmdoc.sty and gmdocc.cls), gmverb.sty, gmutils.sty, gmiflink.sty and also some standard packages: hyperref.sty, color.sty, geometry.sty, multicol.sty, lmodern.sty, fontenc.sty that should be installed on your computer by default.

If you had not installed the mwcls classes (available on CTAN and present in TeX Live e.g.), the result of your compilation might differ a bit from the .pdf provided in this .zip archive in formatting: If you had not installed mwcls, the standard article.cls class would be used.

The Code

```
\@ifpackageloaded{hyperref}{}{\message {^^J^^J gmiflink package:
    There's no use of me without hyperref package, I end my input.^^J}%
          \endinput}
```

```
\providecommand\empty{}
```

A new counter, just in case

GMhlabel

```
\newcounter{GMhlabel}
\setcounter{GMhlabel}{0}
```

The macro given below creates both hypertarget and hyperlabel, so that you may reference both ways: via `\hyperlink` and via `\ref`. It's pattern is the `\label` macro, see LaTeX Source2e, file x, line 32.

But we don't want to gobble spaces before and after. First argument will be a name of the hypertarget, by default the same as typeset text, i.e., argument #2.

\gmhypertarget

```
\DeclareRobustCommand*\gmhypertarget{%
  \@ifnextchar{[}{\gm@hypertarget}{\@dblarg{\gm@hypertarget}}}
```

\gm@hypertarget

```
\def\gm@hypertarget[#1]#2{% If argument #1 = \empty, then we'll use #2, i.e., the
          same as name of hypertarget.
  \refstepcounter{GMhlabel}% we \label{\gmht@firstpar}
  \hypertarget{#1}{#2}%
  \protected@write\@auxout{}{%
    \string\newlabel{#1}{{#2}{\thepage}{\relax}{GMhlabel.\arabic{%
          GMhlabel}}{}}}%
}% end of \gm@hypertartget.
```

We define a macro such that if the target exists, it makes `\ref`, else it typesets ordinary text.

\gmifref

```
\DeclareRobustCommand*\gmifref{\@ifnextchar{[}{\gm@ifref}{% ]
```

```
    \@dblarg{\gm@ifref}}}
```

\gm@ifref

```
\def\gm@ifref[#1]#2{%
  \expandafter\ifx\csname r@#1\endcsname\relax\relax%
  #2\else\ref{#1}\fi%
}% end of \gm@ifref
```

\gmiflink

```
\DeclareRobustCommand*\gmiflink{\@ifnextchar{[}{\gm@iflink}{%
    \@dblarg{\gm@iflink}}}
```

\gm@iflink

```
\def\gm@iflink[#1]#2{%
  \expandafter\ifx\csname r@#1\endcsname\relax\relax%
  #2\else\hyperlink{#1}{#2}\fi%
}% end of \gm@iflink
```

It's robust because when just \newcommand*ed, use of \gmiflink in an indexing macro resulted in errors: \@ifnextchar has to be \noexpanded in \edefs.

```
\endinput
```

The old version — all three were this way primarily.

```
\newcommand*\gmiflink[2][\empty]{{%
  \def\gmht@test{\empty}\def\gmht@firstpar{#1}%
  \ifx\gmht@test\gmht@firstpar\def\gmht@firstpar{#2}\fi%
  \expandafter\ifx\csname r@\gmht@firstpar\endcsname\relax\relax%
  #2\else\hyperlink{\gmht@firstpar}{#2}\fi%
}}
```