

Grzegorz ‘Natorr’ Murzynowski

The gmiflink Package*

Written by Grzegorz ‘Natorr’ Murzynowski,
natorr at o2 dot pl

©2005, 2006 by Grzegorz ‘Natorr’ Murzynowski.

This program is subject to the L^AT_EX Project Public License.

See <http://www.ctan.org/tex-archive/help/Catalogue/licenses.lppl.html> for the details of that license.

LPPL status: "author-maintained".

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{gmiflink}
3 [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* L^AT_EX 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 automatic 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

Just put the `gmiflink.sty` somewhere in the `texmf/tex/latex` branch. Creating a `texmf/tex/latex/gm` directory may be advisable if you consider using other packages written by me.

Then you should refresh your T_EX 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.

```
gmiflink.sty
README
gmiflinkDoc.tex
gmiflinkDoc.pdf
```

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 L^AT_EX 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 T_EX 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

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

6 \providecommand\empty{}
   A new counter, just in case

7 \newcounter{GMhlabel}
8 \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 L^AT_EX 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 9 \DeclareRobustCommand*\gmhypertarget{%
10   \@ifnextchar{[}{\gm@hypertarget}{\@dblarg{\gm@hypertarget}}}
11 \def\gm@hypertarget[#1]#2{% If argument #1 = \empty, then we'll use #2, i.e., the
    same as name of hypertarget.
12   \refstepcounter{GMhlabel}% we \label{\gmht@firstpar}
13   \hypertarget{#1}{#2}%
14   \protected@write\@auxout{}{%
15     \string\newlabel{#1}{#2}{\thepage}{\relax}{GMhlabel.\arabic{
        GMhlabel}}{}}}%
16 }% end of \gm@hypertarget.
```

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

```
\gmifref 17 \DeclareRobustCommand*\gmifref{\@ifnextchar{[}{\gm@ifref}{% ]
18   \@dblarg{\gm@ifref}}}
```

```

19 \def\gm@ifref[#1]#2{%
20   \expandafter\ifx\csname_r@#1\endcsname\relax\relax%
21   #2\else\ref{#1}\fi%
22 }% end of \gm@ifref

\gmiflink 23 \DeclareRobustCommand*\gmiflink{\@ifnextchar{[]{\gm@iflink}{%
24   \@dblarg{\gm@iflink}}}%
25 \def\gm@iflink[#1]#2{%
26   \expandafter\ifx\csname_r@#1\endcsname\relax\relax%
27   #2\else\hyperlink{#1}{#2}\fi%
28 }% 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`.

```
29 \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%
}

```