The undolabl package

H.-Martin Münch
<Martin.Muench at Uni-Bonn.de>

2025-01-28 v1.0o

Abstract

This LaTeX package allows to override existing labels, especially automatically generated ones.

WARNING: Since version 1.0d [2010-07-15] the `\undonewlabel` command takes only one argument,
`\undonewlabel{<label name>}`,
instead of two,
`\undonewlabel{<label name>}{\on@line}`.
Packages or documents, which used older versions of the undolabl package, must be updated by removing the second argument of `\undonewlabel`, i. e. the `{\on@line}`.

Note: The main code of this package was invented by
ULRICH DIEZ
and first published in the news:comp.text.tex newsgroup at
Sun, 20 Apr 2008 16:39:26 +0200, with subject:
Re: How to undefine/overwrite a label? (see e. g. https://groups.google.com/g/comp.text.tex/c/MBiR-EpPceo/m/2pdwkZP-bK8J).
While **ULRICH DIEZ** neither wanted to create a package himself yet nor have one published under his name, he granted the publication of his code. Therefore: Thanks! I submitted this package to CTAN (after some updates of the documentation, mainly layout, typos and such things) and try to maintain it.

Disclaimer for web links: The author is not responsible for any contents referred to in this work unless having full knowledge of illegal contents. If any damage occurs by the use of information presented there, only the author of the respective pages might be liable, not the one who has referred to those pages.

Contents

1 Introduction

The package's name `undolabl` is an eight-letter abbreviation for the phrases "undo" and "label".

This package allows to override existing labels, especially automatically generated ones.

When an instance of the `\label`-macro occurs in the document (e. g. `\label{FOO}`), then a delayed `\write` (a write which is performed at shipout-time when the page-number is determined) to the `.aux` file is issued and you find in the `.aux` file something like:

```
\newlabel{FOO}{{eval(\@currentlabel)}{eval(\thepage)}%
 {eval(\@currentlabelname)}{eval(\@currentHref)}{%
 eval(\@kernel@reserved@label@data)}}
```

(in one line). During the beginning of the next LaTeX-run, the `.aux` file will be read and `\newlabel{FOO}{...}` is expanded to `\@newl@bel r{FOO}{...}`, which in turn gets expanded to something like (pseudo code):

```
IF   (macro \r@FOO is already defined)
THEN - issue an error-message,
     - make sure that the multiply-defined-labels-warning occurs
       in the log-file.
ELSE \def\r@FOO{...}
ENDIF
```

- At the beginning of the LaTeX-run, all the `\r@<label>`-macros get defined from reading the `.aux` file. The `\r@<label>`-macros get used by the referencing-macros (`\ref`, `\pageref`,...) during the LaTeX-run.
- During the LaTeX-run, the `.aux` file gets rewritten.
- At the end of the LaTeX-run, the `.aux` file (which was rewritten/newly created during the LaTeX-run) is read in order to detect whether references have changed during the current LaTeX-run.

But this time `\@newl@bel` is redefined (`\let` equal to `\@testdef`) and thus this time
`\@newl@bel r{FOO}{{...}{...}{...}{...}{...}}`
expands to something like (pseudo code):

```
Compare the (newly written) third argument (that is:
{{...}{...}{...}{...}{...}})
to the (current/former) definition of \r@FOO.
If the two are different, then some page- or section-number
related to referencing has changed from the last to the current
LaTeX-run, thus in this case issue a message in the log-file:
"References may have changed. Rerun LaTeX in order to get
cross-references right".
```

So what do you need to do in order to override a label:

- First you need to write to the `.aux` file to silently undefine the associated `\r@<label>`-macro if it is already defined. That is why `\overridelabel` writes in terms of `\protected@write` to the `.aux` file:

  ```
  \undonewlabel{<label>}
  ```

  `\undonewlabel` "undefines" the `\r@<label>`-macro. (How this works will be explained below.)
- Then `\overridelabel` can call `\label{<label>}` again and thus produce another `\newlabel{<label>}`-entry to the `.aux` file.

In the `.aux` file all this results in a sequence like:

```
% from the former \label-call:
\newlabel{<label>}{...}
|->| \r@<label>-macro gets produced.
% from the \overrridelabel-call:
% - call to \undonewlabel within \overrridelabel:
\undonewlabel{<label>}
|->| \r@<label>-macro gets destroyed.
% - call to \label within \overrridelabel:
\newlabel{<label>}{...}
|->| a new \r@<label>-macro but no multiply-label-defined warning
gets produced.
```

There is another issue left:

- It was said that `.aux` file is read at the beginning and at the end of the LaTeX-run for detecting whether references have changed.
- When overriding a `<label>`, there will be several `\newlabel{<label>}`-calls associated to the same label-name in the `.aux` file.
- At the beginning of the LaTeX-run only the last one counts for defining the associated `\r@<label>`-macro.
- But at the end, when the new `.aux` file is read, they all count and thus with all these entries but the last one, the above-mentioned `\@testdef`-comparison will yield difference and thus in any case cause a warning-message about references having changed although that might not be a correct statement.

ULRICH DIEZ decided to catch this up by his `\undolabl@testdef` command. The `\undolabl@testdef`-comparison-mechanism gets enhanced via "replacing". This works as follows: When the `.aux` file is read at the beginning of the LaTeX-run, `\@newl@bel` is not let equal to `\@testdef`. When the `.aux` file is read at the end of the LaTeX-run, `\@newl@bel` is let equal to `\@testdef`. Thus it is sufficient to write into the beginning of the `.aux` file a direction which leads to `\let`ting `\@newl@bel` equal to `\undolabl@testdef` in case its definition equals `\@testdef`. That direction is called "`\reset@newl@bel`". Also, when the `.aux` file is read at the end of the LaTeX-run, `\undonewlabel`-entries therein should do nothing, thus `\undonewlabel` is `\let` equal to `\@gobble`.

2 Usage

Load the package placing

```
\usepackage{undolabl}
```

in the preamble of your LaTeX 2_ε source file.

When an existing label shall be replaced by a new one, say
`\overridelabel{<label name>}`
(where `<label name>` is the name of the label to be replaced by the new one), instead of just `\label{<label name>}`, which would produce a
`LaTeX Warning: Label ‘<label name>’ multiply defined.`

3 Example

```
⟨*example⟩
\documentclass[british]{article}[2024/06/29]% v1.4n
\usepackage{undolabl}[2025-01-28]% v1.0o
%% There are no options for the undolabl package.
\usepackage[%
 extension=pdf,%
 plainpages=false,%
 pdfpagelabels=true,%
 hyperindex=false,%
 pdflang={en},%
 pdftitle={undolabl package example},%
 pdfauthor={H.-Martin Muench, after Ulrich Diez},%
 pdfsubject={Example for the undolabl package},%
 pdfkeywords={LaTeX, undolabl, undolabel},%
 pdfview=FitH,%
 pdfstartview=FitH,%
 pdfpagelayout=OneColumn,%
 bookmarksopen=true%
]{hyperref}[2024-11-05]% v7.01l; when you want to use nameref
\listfiles
\begin{document}
\pagenumbering{arabic}
\section*{Example for undolabl}
This example demonstrates the use of package\newline
\textsf{undolabl}, v1.0o as of 2025-01-28 (HMM).\newline
For details please see the documentation!
\bigskip

\section{Test}
text \label{testlabel} and text \overridelabel{testlabel}\par
page-reference: \pageref{testlabel}\par%%  -> page 4
sectional-reference: \ref{testlabel}\par%% -> section 4
name-reference: \nameref{testlabel}%%      -> Still another test

\newpage
\section{Another test}
text \overridelabel{testlabel}\par
page-reference: \pageref{testlabel}\par%%  -> page 4
sectional-reference: \ref{testlabel}\par%% -> section 4
name-reference: \nameref{testlabel}%%      -> Still another test

\newpage
\section{Yet another test}
text \overridelabel{testlabel}\par
page-reference: \pageref{testlabel}\par%%  -> page 4
sectional-reference: \ref{testlabel}\par%% -> section 4
name-reference: \nameref{testlabel}%%      -> Still another test

\newpage
\section{Still another test}
text \overridelabel{testlabel}\par
page-reference: \pageref{testlabel}\par%%  -> page 4
sectional-reference: \ref{testlabel}\par%% -> section 4
name-reference: \nameref{testlabel}%%      -> Still another test
\end{document}
⟨/example⟩
```

4 The implementation

For a somewhat longer description see section 1.

We start off by checking that we are loading into LaTeX 2_ε and announcing the name and version of this package.

```
⟨*package⟩
\NeedsTeXFormat{LaTeX2e}[2024-11-01]
\ProvidesPackage{undolabl}[2025-01-28 v1.0o Overriding labels (HMM)]

\@ifl@t@r\fmtversion{2024/11/01}{}{\PackageError{undolabl}{%
  LaTeX format 2024-11-01 or newer needed}{%
  Needed LaTeX format version: 2024-11-01 or newer.\MessageBreak%
  Found\space\space LaTeX format version: \fmtversion.\MessageBreak%
  Please update your TeX distribution!\MessageBreak%
  Loading of undolabl package is aborted.}
  \expandafter\endinput}

```

\overridelabel In order to override a label, first one needs to write to the `.aux` file to silently undefine the associated `\r@<label>`-macro if it is already defined. That is why `\overridelabel` writes in terms of `\protected@write` to the `.aux` file: `\undonewlabel{<label>}`, "undefining" the `\r@<label>`-macro. And a message about this is given.

```
\newcommand\overridelabel[1]{%
\@bsphack%
  \if@filesw%
    \protected@write\@auxout{}{\string\undonewlabel{#1}}%
    \@overriddenmessage s{#1}%
```

Then `\overridelabel` can call `\label{<label>}` again and thus produce another `\newlabel{<label>}`-entry to the `.aux` file.

```
    \label{#1}%
  \fi%
\@esphack%
}

```

\undonewlabel

```
\newcommand\undonewlabel{\@und@newl@bel r}

```

\@und@newl@bel If `\r@<label>` is undefined, give an error message:

```
\newcommand\@und@newl@bel[2]{%
  \@ifundefined{#1@#2}{%
    \PackageError{undolabl}{Label `#2' shall be overridden ^^J%
      although it does not yet exist}{%
      A label which does not exist cannot be overridden.}%
```

otherwise:
Undefine `\r@label` via letting it equal to `\relax`:

```
  }{\expandafter\global
    \expandafter\let
    \csname #1@#2\endcsname\relax%
   }%
  }

```

\@overriddenmessage Command for the notification of overriding a label:

```
\newcommand\@overriddenmessage[2]{%
  \@ifundefined{#1@#2}{\global\@namedef{#1@#2}{i}}{%
    \expandafter\g@addto@macro\csname #1@#2\endcsname{i}%
    }%
  \PackageNote{undolabl}{Label `#2' overridden}%
  }

```

\undolabl@testdef

```
\newcommand\undolabl@testdef[3]{%
  \@ifundefined{s@#2}\@secondoftwo\@firstofone{%
    \expandafter\ifx\csname s@#2\endcsname\empty
      \expandafter\@firstofone
    \else%
    \expandafter\xdef\csname s@#2\endcsname{%
      \expandafter\expandafter
      \expandafter\@gobble
      \csname s@#2\endcsname
      }%
    \expandafter\@gobble
    \fi%
   }{\@testdef{#1}{#2}{#3}%
    }%
  }

```

\reset@newl@bel

```
\newcommand\reset@newl@bel{%
  \ifx\@newl@bel\@testdef%
    \let\@newl@bel\undolabl@testdef%
    \let\undonewlabel\@gobble%
  \fi%
  }

```

AddToHook{begindocument} At `begindocument` it is checked whether writing to files is allowed. Some packages (e. g. tikz and selectp) sometimes prevent the output to the `aux` file. In that case a warning or an error message is issued. This is no problem as long as there is/was another compilation run where the labels can/could be processed via the `aux` file.

If writing is allowed, a `\reset@newl@bel` (see above) is written into the `aux` file.

```
\AddToHook{begindocument}{%
  \if@filesw%
    \immediate\write\@auxout{\string\reset@newl@bel}%
```

When writing to files is not allowed, nothing can be done. But when the labels were already processed via the `aux` file, nothing needs to be done (if enough compilation runs have been done before).

```
  \else%
    \IfPackageLoadedTF{tikz}{\PackageWarningNoLine{undolabl}{%
      The undolabl package was not allowed to write to an\MessageBreak%
      .aux file. This package does not work without access\MessageBreak%
      to an .aux file.\MessageBreak%
      It is OK if the .aux file was already updated\MessageBreak%
      by a previous compiler run\MessageBreak%
      and would not have changed anyway.}%
    }{\PackageError{undolabl}{No writing to auxiliary file allowed}{%
        The undolabl package was not allowed to write to an .aux file.\MessageBreak%
        This package does not work without access to an .aux file.\MessageBreak%
        Press Ctrl+Z to exit.\MessageBreak%
        But it is OK if the .aux file was already updated\MessageBreak%
        by a previous compiler run\MessageBreak%
```

```
        and would not have changed anyway.}%
    }%
\fi%
```

The undolabl and the acronym packages shared some commands with the same names and where therefore incompatible. Jan Heisswolf (ITIV at KIT) reported this in 2014 (Thanks!). Changing the macros in the undolabl package also required changing them in the pageslts package and providing a fallback mechanism for the old commands used in existing `.aux` files. When I wanted to implement this in 2015, it turned out the acronym package had been updated a week before (Thanks!) and fixed the incompatibility. Now what remains to be done is just to check that no ancient acronym package is used:

```
\IfPackageLoadedT{acronym}{\IfPackageAtLeastF{acronym}{2015/03/21}{%
  \PackageError{undolabl}{Incompatible old acronym package detected}{%
    The undolabl package is not compatible with acronym package\MessageBreak%
    older than 2015/03/21 v1.41.\MessageBreak%
    Found version: \csname ver@acronym.sty\endcsname .\MessageBreak%
    Please update your acronym package!%
}}}}
```

⟨/package⟩

5 Installation

5.1 Downloads

Everything is available at https://www.ctan.org but may need additional packages themselves.

`undolabl.dtx` For unpacking the `undolabl.dtx` file and constructing the documentation it is required:

- TEX Format LATEX 2ε [2024-11-01] (or newer), https://www.CTAN.org
- document class ltxdoc, 2024/02/08, v2.1j, https://www.ctan.org/pkg/ltxdoc
- package holtxdoc, 2019/12/09, v0.30, https://www.ctan.org/pkg/holtxdoc
- package hypdoc, 2023-10-26, v1.19, https://www.ctan.org/pkg/hypdoc

`undolabl.sty` The `undolabl.sty` for LATEX 2ε (i. e. each document using the undolabl package) requires:

- TEX Format LATEX 2ε [2024-11-01] (or newer), https://www.CTAN.org.

`undolabl-example.tex` The `undolabl-example.tex` requires the same files as all documents using the undolabl package and additionally:

- class article, 2024/06/29, v1.4n, https://www.ctan.org/pkg/article
- package hyperref, 2024-11-05, v7.01, https://www.ctan.org/pkg/hyperref, if `\nameref` shall be used (and when the references shall be hyperlinked, of course)
- package undolabl, 2025-01-28, v1.0o, https://www.ctan.org/pkg/undolabl (Well, it is the example file for this package, and because you are reading the documentation for the undolabl package, it can be assumed that you already have some version of it – is it the current one?)

`Münch` A hyperlinked list of my (other) packages can be found at https://www.ctan.org/author/muench-hm.

5.2 Package, unpacking TDS

Package. This package is available at https://www.ctan.org/pkg/undolabl, especially the manual (README, undolabl.pdf), the example (source: undolabl-example.tex, compiled: undolabl-example.pdf), and the undolabl.dtx. There is also an undolabl.tds.zip available: https://mirror.ctan.org/install/macros/latex/contrib/undolabl.tds.zip (everything in TDS compliant, compiled format), which additionally contains

undolabl.ins	The installation file.
undolabl.drv	The driver to generate the documentation.
undolabl.sty	The .style file.
undolabl-example.tex	The example file.

For required other packages, please see the preceding subsection.

Unpacking. The .dtx file is a self-extracting docstrip archive. The files are extracted by running the ..dtx through plain TEX:

```
tex undolabl.dtx
```

About generating the documentation see paragraph 5.4 below.

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as texmf tree):

```
undolabl.sty          → tex/latex/undolabl/undolabl.sty
undolabl.pdf          → doc/latex/undolabl/undolabl.pdf
undolabl-example.tex  → doc/latex/undolabl/undolabl-example.tex
undolabl-example.pdf  → doc/latex/undolabl/undolabl-example.pdf
undolabl.dtx          → source/latex/undolabl/undolabl.dtx
```

If you have a docstrip.cfg that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

5.3 Refresh file name databases

If your TEX distribution (TEX Live, MiKTEX, ...) relies on file name databases, you must refresh these. For example, TEX Live users run texhash or mktexlsr.

5.4 Some details for the interested

Unpacking with LATEX. The .dtx chooses its action depending on the format:

plain TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{undolabl.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by a configuration file `ltxdoc.cfg`. For instance, put the following line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLaTeX:

```
pdflatex undolabl.dtx
makeindex -s gind.ist undolabl.idx
pdflatex undolabl.dtx
makeindex -s gind.ist undolabl.idx
pdflatex undolabl.dtx
```

6 Acknowledgements

The main code of this package was invented by
ULRICH DIEZ (eu_angelion@web.de)
and first published in the news:comp.text.tex newsgroup at
Sun, 20 Apr 2008 16:39:26 +0200, with subject:
Re: How to undefine/overwrite a label? (see e. g. https://groups.google.com/g/comp.text.tex/c/MBiR-EpPceo/m/2pdwkZP-bK8J as well as https://groups.google.com/g/comp.text.tex/c/0jzpQsn5dPs/m/Lrb45ByCZtMJ).
I (H.-Martin Münch) would like to thank **ULRICH DIEZ** for this as well as for his permission to publish it on CTAN as well as for his bug reports. I also thank those anonymous people who had published the package somewhere else on the internet, where I found it first. Further I would like to thank Heiko Oberdiek for providing a lot (!) of useful packages (from which I also got everything I know about creating a file in `dtx` format, OK, say it: copying).

7 History

[2008/04/20 v0.3(a)]

- Created by **ULRICH DIEZ**.

[2010/04/08 v0.3b]

- `.dtx` updated by H.-Martin Münch, submitted to CTAN (no changes in the style code).

[2010/06/01 v1.0(a)]

- `.dtx` updated: some minor corrections in the documentation, an internal renaming for possible better compatibility with other packages.
- The main code of this package was invented in 2008 by **ULRICH DIEZ** (eu_angelion@web.de) and published on the internet. Because **ULRICH DIEZ** neither wanted to create a package himself yet nor have one published under his name, but granted the publication of his code (Thanks!), I had to change author/maintainer of this package and resubmit it.

[2010/06/03 v1.0b]

- Found an unchanged reference to the package authors/maintainer.
- Example adapted to other examples of mine.

- Updated references to other packages.
- TDS locations updated.
- Several changes in the documentation and the README file.

[2010/06/24 v1.0c]

- holtxdoc warning in drv updated.
- Corrected the location of the package at CTAN. (TDS of this version was still missing due to a packaging error.)
- Updated references to other packages: hyperref and pagesLTS (which has been renamed to pageslts and is no longer referenced since v1.0h).

[2010/07/15 v1.0d]

- There was another update by **ULRICH DIEZ** on news:comp.text.tex at Mon, 21 Apr 2008 23:04:03 +0200, see e. g. https://groups.google.com/g/comp.text.tex/c/MBiR-EpPceo/m/2pdwkZP-bK8J, which now has been included in this package.
- Put more emphasis on **ULRICH DIEZ** writing the initial code.
- Updated references to other packages and corrected the given location of the undolabl.tds.zip file at CTAN.org.

[2010/07/25 v1.0e]

- Bugs reported by **ULRICH DIEZ** on news:comp.text.tex at Sat, 17 Jul 2010 12:27:10 +0200, subject Re: CTAN Update: undolabl, see e. g. https://groups.google.com/g/comp.text.tex/c/0jzpQsn5dPs/m/Lrb45ByCZtMJ, eradicated.
- \StopEventually added and \CheckSum value corrected (was 0).
- Minor details.

[2010/07/29 v1.0f]

- Corrected diverse urls, updated references to other packages.

[2010/09/12 v1.0g]

- There was a wrong % behind 2010/07/29 v1.0f, resulting in the version being displayed as "v1.0fOverriding".
- Changed the \unit definition (got rid of an old \rm).
- A lot of small changes.

[2011/02/01 v1.0h]

- Updated to new version of the hyperref package.
- Removed /muench/ from the path at diverse locations.
- Some small changes.

[2011/06/26 v1.0i]

- The holtxdoc package was fixed (recent: 2011/02/04, v0.21), therefore the warning in `drv` could be removed. – Adapted the style of this documentation to new OBERDIEK `dtx` style.
- There is a new version of the used hyperref package.
- Quite some changes in the `.dtx`/documentation.

[2011/08/08 v1.0j]

- The pagesLTS package has been renamed to pageslts: 2011/08/08, v1.2a.
- Some minor changes.

[2012/01/01 v1.0k]

- Bugfix: Obsolete installation path given in the documentation, updated.
- Bugfix: A section was broken in the documentation (text had been lost but was recovered now).
- Update of documentation, README, and `dtx` internals.

[2015/03/29 v1.0l]

- Incompatibility with acronym package fixed – by acronym package, thanks! Check for older versions introduced.
- Introduced a check, whether writing to the `aux` file is allowed.
- Update of documentation, README, and `dtx` internals.

[2023-02-14 v1.0m]

- Removed `\unit`.
- Converted to UTF-8.
- Updated to LaTeX format 2022-11-01.
- Documentation updated (there were lots of outdated urls).

[2024-12-05 v1.0n]

- In an edge case, `\reset@newl@bel` was issued too late. Fixed.
- Update of minor details in the code and update of documentation (e. g. regarding of the number of arguments in labels).

[2025-01-28 v1.0o]

- Documentation update.

When you find a mistake or have a suggestion for an improvement of this package, please send an e-mail to the maintainer, thanks! (Please see BUG REPORTS in the README.)

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; plain numbers refer to the code lines where the entry is used.