

Marking up differences between latex files with *latexdiff*

F.J. Tilmann*

October 19, 2007

Preamble

latexdiff is a Perl script, which compares two latex files and marks up significant differences between them. Various options are available for visual markup using standard latex packages such as *color.sty*. Changes not directly affecting visible text, for example in formatting commands, are still marked in the latex source. A rudimentary revision facility is provided by another Perl script, *latexrevise*, which accepts or rejects all changes. Manual editing of the difference file can be used to override this default behaviour and accept or reject selected changes only.

There is no explicit support for annotations as these are trivial to implement. For example, I include the following command definition in the preamble

```
\newcommand{\remark}[1]{\bf [ \footnotesize #1 ]}
```

and mark up annotations as follows

```
... The roadrunner is the fastest running bird \remark{Check this  
again with a zoologist!}. The most famous roadrunner ...
```

Alternatively, instead of a command like `\remark` in the example just given, an equivalent annotation environment could be defined. *latexrevise* can remove such comments or environments from the text body.

It is planned that the revision capabilities of this system will be further expanded, dependent on the amount of feedback received.

On the following pages you find the *man* pages for *latexdiff* and *latexrevise* and a simple example.

*tilmann@esc.cam.ac.uk

1 NAME

latexdiff - determine and markup differences between two latex files

2 SYNOPSIS

latexdiff [OPTIONS] *old.tex new.tex > diff.tex*

3 DESCRIPTION

Briefly, *latexdiff* is a utility program to aid in the management of revisions of latex documents. It compares two valid latex files, here called `old.tex` and `new.tex`, finds significant differences between them (i.e., ignoring the number of white spaces and position of line breaks), and adds special commands to highlight the differences. Where visual highlighting is not possible, e.g. for changes in the formatting, the differences are nevertheless marked up in the source.

The program treats the preamble differently from the main document. Differences between the preambles are found using line-based differencing (similarly to the Unix diff command, but ignoring white spaces). A comment, "%DIF >" is appended to each added line, i.e. a line present in `new.tex` but not in `old.tex`. Discarded lines are deactivated by prepending "%DIF <". Changed blocks are preceded by comment lines giving information about line numbers in the original files. Where there are insignificant differences, the resulting file `diff.tex` will be similar to `new.tex`. At the end of the preamble, the definitions for *latexdiff* markup commands are inserted. In differencing the main body of the text, *latexdiff* attempts to satisfy the following guidelines (in order of priority):

1. 1

If both `old.tex` and `new.tex` are valid LaTeX, then the resulting `diff.tex` should also be valid LaTeX. (NB If a few plain TeX commands are used within `old.tex` or `new.tex` then `diff.tex` is not guaranteed to work but usually will).

2. 2

Significant differences are determined on the level of individual words. All significant differences, including differences between comments should be clearly marked in the resulting source code `diff.tex`.

3. 3

If a changed passage contains text or text-producing commands, then running `diff.tex` through LaTeX should produce output where added and discarded passages are highlighted.

4. 4

Where there are insignificant differences, e.g. in the positioning of line breaks, `diff.tex` should follow the formatting of `new.tex`

For differencing the same algorithm as *diff* is used but words instead of lines are compared. An attempt is made to recognize blocks which are completely changed such that they can be marked up as a unit. Comments are differenced line by line but the number of spaces within comments is ignored. Commands including all their arguments are generally compared as one unit, i.e., no markup is inserted into the arguments of commands. However, for a selected number of commands (for example, `\caption` and all sectioning commands) the last argument is known to be text. This text is split into words and differenced just as ordinary text (use options to show and change the list of text commands, see below). As the algorithm has no detailed knowledge of LaTeX, it assumes all pairs of curly braces immediately following a command (i.e. a sequence of letters beginning with a backslash) are arguments for that command. As a restriction to condition 1 above it is thus necessary to surround all arguments with curly braces, and to not insert extraneous spaces. For example, write

```
\section{\textem{This is an emphasized section title}}
```

and not

```
\section {\textem{This is an emphasized section title}}
```

or

```
\section\textem{This is an emphasized section title}
```

even though all varieties are the same to LaTeX (but see `--allow-spaces` option which allows the second variety).

For environments whose content does not conform to standard LaTeX or where graphical markup does not make sense all markup commands can be removed by setting the `PICTUREENV` configuration variable, set by default to `picture` and `DIFnomarkup` environments; see `--config` option). The latter environment (`DIFnomarkup`) can be used to protect parts of the latex file where the markup results in illegal markup. You have to surround the offending passage in both the old and new file by `\begin{DIFnomarkup}` and `\end{DIFnomarkup}`. You must define the environment in the preambles of both old and new documents. I prefer to define it as a null-environment,

```
\newenvironment{DIFnomarkup}{}{}
```

but the choice is yours. Any markup within the environment will be removed, and generally everything within the environment will just be taken from the new file.

It is also possible to difference files which do not have a preamble. In this case, the file is processed in the main document mode, but the definitions of the markup commands are not inserted.

All markup commands inserted by *latexdiff* begin with `"\DIF"`. Added blocks containing words, commands or comments which are in `new.tex` but not in `old.tex` are marked by `\DIFaddbegin` and `\DIFaddend`. Discarded blocks are marked by `\DIFdelbegin` and `\DIFdelend`. Within added blocks all text is highlighted with `\DIFadd` like this: `\DIFadd{Added text block}` Selected ‘safe’ commands can be contained in these text blocks as well (use options to show and change the list of safe commands, see below). All other commands as well as braces `"{"` and `"}"` are never put within the scope of `\DIFadd`. Added comments are marked by prepending `"%DIF > "`.

Within deleted blocks text is highlighted with `\DIFdel`. Deleted comments are marked by prepending `"%DIF < "`. Non-safe command and curly braces within deleted blocks are commented out with `"%DIFDEL CMD < "`.

4 OPTIONS

Preamble

The following options determine the visual markup style by adding the appropriate command definitions to the preamble. See the end of this section for a description of available styles.

--type=markupstyle or -t markupstyle

Add code to preamble for selected markup style. This option defines `\DIFadd` and `\DIFdel` commands. Available styles:

UNDERLINE CTRADITIONAL TRADITIONAL CFONT FONTSTRIKE INVISIBLE
CHANGEBAR CCHANGEBAR CULINECHBAR CFONTCBHBAR

[Default: UNDERLINE]

--subtype=markstyle or -s markstyle

Add code to preamble for selected style for bracketing commands (e.g. to mark changes in margin). This option defines `\DIFaddbegin`, `\DIFaddend`, `\DIFdelbegin` and `\DIFdelend` commands. Available styles: SAFE MARGINAL DVIPSCOL

[Default: SAFE]

--floatype=markstyle or -f markstyle

Add code to preamble for selected style which replace standard marking and markup commands within floats (e.g., marginal remarks cause an error within floats so marginal marking can be disabled thus). This option defines all `\DIF...FL` commands. Available styles: FLOATSAFE TRADITIONALSAFE IDENTICAL

[Default: FLOATSAFE]

--encoding=enc or -e enc

Specify encoding of old.tex and new.tex. Typical encodings are `ascii`, `utf8`, `latin1`, `latin9`. A list of available encodings can be obtained by executing

```
perl -MEncode -e 'print join ("\n",Encode-encodings("all"));' >
```

[Default encoding is `utf8` unless the first few lines of the preamble contain an invocation `\usepackage[.]{inputenc}` in which case the encoding chosen by this command is assumed. Note that ASCII (standard latex) is a subset of `utf8`]

--preamble=file or -p file

Insert file at end of preamble instead of generating preamble. The preamble must define the following commands `\DIFaddbegin`, `\DIFaddend`, `\DIFadd{.}`, `\DIFdelbegin`, `\DIFdelend`, `\DIFdel{.}`, and varieties for use within floats `\DIFaddbeginFL`, `\DIFaddendFL`, `\DIFaddFL{.}`, `\DIFdelbeginFL`, `\DIFdelendFL`, `\DIFdelFL{.}` (If this option is set `-t`, `-s`, and `-f` options are ignored.)

--packages=pkg1,pkg2,..

Tell latexdiff that .tex file is processed with the packages in list loaded. This is normally not necessary if the .tex file includes the preamble, as the preamble is automatically scanned for `\usepackage` commands. Use of the **--packages** option disables automatic scanning, so if for any reason package specific parsing needs to be switched off, use **--packages=none**. The following packages trigger special behaviour:

amsmath

Configuration variable `amsmath` is set to `align*` (Default: `eqnarray*`)

endfloat

Ensure that `\begin{figure}` and `\end{figure}` always appear by themselves on a line.

hyperref

Change name of `\DIFadd` and `\DIFdel` commands to `\DIFaddtex` and `\DIFdeltex` and define new `\DIFadd` and `\DIFdel` commands, which provide a wrapper for these commands, using them for the text but not for the link defining command (where any markup would cause errors).

[Default: scan the preamble for `\usepackage` commands to determine loaded packages.]

--show-preamble

Print generated or included preamble commands to stdout.

Configuration

--exclude-safecmd=exclude-file or **-A exclude-file** or **--exclude-safecmd="cmd1,cmd2,..."**

--replace-safecmd=replace-file

--append-safecmd=append-file or **-a append-file** or **--append-safecmd="cmd1,cmd2,..."**

Exclude from, replace or append to the list of regular expressions (RegEx) matching commands which are safe to use within the scope of a `\DIFadd` or `\DIFdel` command. The file must contain one Perl-RegEx per line (Comment lines beginning with `#` or `%` are ignored). Note that the RegEx needs to match the whole of the token, i.e., `/^regex$/` is implied and that the initial `"\"` of the command is not included. The **--exclude-safecmd** and **--append-safecmd** options can be combined with the **---replace-safecmd** option and can be used repeatedly to add cumulatively to the lists. **--exclude-safecmd** and **--append-safecmd** can also take a comma separated list as input. If a comma for one of the regex is required, escape it thus `"\"`. In most cases it will be necessary to protect the comma-separated list from the shell by putting it in quotation marks.

--exclude-textcmd=exclude-file or **-X exclude-file** or **--exclude-textcmd="cmd1,cmd2,..."**

--replace-textcmd=replace-file

--append-textcmd=append-file or **-x append-file** or **--append-textcmd="cmd1,cmd2,..."**

Exclude from, replace or append to the list of regular expressions matching commands whose last argument is text. See entry for **--exclude-safecmd** directly above for further details.

--replace-context1cmd=replace-file

--append-context1cmd=append-file or **=item --append-context1cmd="cmd1,cmd2,..."**

Replace or append to the list of regex matching commands whose last argument is text but which require a particular context to work, e.g. `\caption` will only work within a figure or table. These commands behave like text commands, except when they occur in a deleted section, when they are disabled, but their argument is shown as deleted text.

--replace-context1cmd=replace-file

--append-context2cmd=append-file or **=item --append-context2cmd="cmd1,cmd2,..."**

As corresponding commands for `context1`. The only difference is that `context2` commands are completely disabled in deleted sections, including their arguments.

--config var1=val1,var2=val2,... or **-c var1=val1,..**

-c configfile

Set configuration variables. The option can be repeated to set different variables (as an alternative to the comma-separated list). Available variables (see below for further explanations):

MINWORDSBLOCK (integer)

FLOATENV (RegEx)

PICTUREENV (RegEx)

MATHENV (RegEx)

MATHREPL (String)

MATHARRENV (RegEx)

MATHARRREPL (String)

ARRENV (RegEx)

COUNTERCMD (RegEx)

--show-safecmd

Print list of RegEx matching and excluding safe commands.

--show-textcmd

Print list of RegEx matching and excluding commands with text argument.

--show-config

Show values of configuration variables.

--show-all

Combine all --show commands.

NB For all --show commands, no `old.tex` or `new.tex` file needs to be specified, and no differencing takes place.

Miscellaneous

--verbose or -V

Output various status information to stderr during processing. Default is to work silently.

--ignore-warnings

Suppress warnings about inconsistencies in length between input and parsed strings and missing characters. These warning messages are often related to non-standard latex or latex constructions with a syntax unknown to `latexdiff` but the resulting difference argument is often fully functional anyway, particularly if the non-standard latex only occurs in parts of the text which have not changed.

--allow-spaces

Allow spaces between bracketed or braced arguments to commands. Note that this option might have undesirable side effects (unrelated scope might get lumped with preceding commands) so should only be used if the default produces erroneous results. (Default requires arguments to directly follow each other without intervening spaces).

--flatten

Replace `\input` and `\include` commands within body by the content of the files in their argument. If `\includeonly` is present in the preamble, only those files are expanded into the document. However, no recursion is done, i.e. `\input` and `\include` commands within included sections are not expanded. The included files are assumed to be located in the same directories as the old and new master files, respectively, making it possible to organise files into old and new directories.

Use of this option is not recommended primarily the processing time for the large documents is prohibitive, and the resulting difference document no longer reflects the structure of the input documents.

--help or -h

Show help text

--version

Show version number

Predefined styles

Major types

The major type determine the markup of plain text and some selected latex commands outside floats by defining the markup commands `\DIFadd{...}` and `\DIFdel{...}`.

UNDERLINE

Added text is wavy-underlined and blue, discarded text is struck out and red (Requires color and ulem packages). Overstriking does not work in displayed math equations such that deleted parts of equation are underlined, not struck out (this is a shortcoming inherent to the ulem package).

CTRADITIONAL

Added text is blue and set in sans-serif, and a red footnote is created for each discarded piece of text. (Requires color package)

TRADITIONAL

Like **CTRADITIONAL** but without the use of color.

CFONT

Added text is blue and set in sans-serif, and discarded text is red and very small size.

FONTSTRIKE

Added text is set in sans-serif, discarded text small and struck out

CCHANGEBAR

Added text is blue, and discarded text is red. Additionally, the changed text is marked with a bar in the margin (Requires color and changebar packages).

CFONTCHBAR

Like CFONT but with additional changebars (Requires color and changebar packages).

CULINECHBAR

Like UNDERLINE but with additional changebars (Requires color, ulem and changebar packages).

CHANGEBAR

No mark up of text, but mark margins with changebars (Requires changebar package).

INVISIBLE

No visible markup (but generic markup commands will still be inserted).

Subtypes

The subtype defines the commands that are inserted at the begin and end of added or discarded blocks, irrespectively of whether these blocks contain text or commands (Defined commands: `\DIFaddbegin`, `\DIFaddend`, `\DIFdelbegin`, `\DIFdelend`)

SAFE

No additional markup (Recommended choice)

MARGIN

Mark beginning and end of changed blocks with symbols in the margin nearby (using the standard `\marginpar` command - note that this sometimes moves somewhat from the intended position.

DVIPSCOL

An alternative way of marking added passages in blue, and deleted ones in red. Note that DVIPSCOL only works with the dvips converter, e.g. not pdflatex. (it is recommended to use instead the main types to effect colored markup, although in some cases coloring with dvipscol can be more complete).

Float Types

Some of the markup used in the main text might cause problems when used within floats (e.g. figures or tables). For this reason alternative versions of all markup commands are used within floats. The float type defines these alternative commands.

FLOATSAFE

Use identical markup for text as in the main body, but set all commands marking the begin and end of changed blocks to null-commands. You have to choose this float type if your subtype is `MARGIN` as `\marginpar` does not work properly within floats.

TRADITIONALSAFE

Mark additions the same way as in the main text. Deleted environments are marked by angular brackets `\[` and `\]` and the deleted text is set in `scriptscript` size. This float type should always be used with the `TRADITIONAL` and `CTRADITIONAL` markup types as the `\footnote` command does not work properly in floating environments.

IDENTICAL

Make no difference between the main text and floats.

Configuration Variables

MINWORDSBLOCK

Minimum number of tokens required to form an independent block. This value is used in the algorithm to detect changes of complete blocks by merging identical text parts of less than `MINWORDSBLOCK` to the preceding added and discarded parts.

[Default: 3]

FLOATENV

Environments whose name matches the regular expression in `FLOATENV` are considered floats. Within these environments, the *latexdiff* markup commands are replaced by their FL varieties.

[Default: `(?:figure|table|plate) [\w\d*@]*`]

PICTUREENV

Within environments whose name matches the regular expression in `PICTUREENV` all *latexdiff* markup is removed (in pathologic cases this might lead to inconsistent markup but this situation should be rare).

[Default: `(?:picture|DIFnomarkup) [\w\d*@]*`]

MATHENV,MATHREPL

If both `\begin` and `\end` for a math environment (environment name matching `MATHENV` or `\[` and `\]`) are within the same deleted block, they are replaced by a `\begin` and `\end` commands for `MATHREPL` rather than being commented out.

[Default: `MATHENV=(?:displaymath|equation)`, `MATHREPL=displaymath`]

MATHARRENV,MATHARRREPL

as `MATHENV,MATHREPL` but for equation arrays

[Default: `MATHARRENV=eqnarray*?`, `MATHREPL=eqnarray`]

ARRENV

If a match to `ARRENV` is found within an inline math environment within a deleted or added block, then the inlined math is surrounded by `\mbox{...}`. This is necessary as underlining does not work within inlined array environments.

[Default: `ARRENV=(?:array|[pbvBV]matrix)`]

COUNTERCMD

If a command in a deleted block which is also in the `textcmd` list matches `COUNTERCMD` then an additional command `\addtocounter{cntcmd}{-1}`, where `cntcmd` is the matching command, is appended in the diff file such that the numbering in the diff file remains synchronized with the numbering in the new file.

[Default: `COUNTERCMD=(?:footnote|part|section|subsection ... |subsubsection|paragraph|subparagraph)`]

5 BUGS

UTF-8 support requires a relatively new version of perl (5.8.0 is sufficient but 5.6.2 up would work OK, too).

Option `allow-spaces` not implemented entirely consistently. It breaks the rules that number and type of white space does not matter, as different numbers of inter-argument spaces are treated as significant.

Please send bug reports to tilmann@esc.cam.ac.uk. Include the serial number of `latexdiff` (from comments at the top of the source or use `--version`). If you come across latex files that are error-free and conform to the specifications set out above, and whose differencing still does not result in error-free latex, please send me those files, ideally edited to only contain the offending passage as long as that still reproduces the problem.

6 SEE ALSO

latexrevise

7 PORTABILITY

latexdiff does not make use of external commands and thus should run on any platform supporting Perl 5.6 or higher. If files with encodings other than ASCII or UTF-8 are processed, Perl 5.8 or higher is required.

The standard version of *latexdiff* requires installation of the Perl package `Algorithm::Diff` (available from *www.cpan.org* - <http://search.cpan.org/~nedkonz/Algorithm-Diff-1.15>) but a stand-alone version, *latexdiff-so*, which has this package inlined, is available, too. *latexdiff-fast* requires the *diff* command to be present.

8 AUTHOR

Copyright (C) 2004-2007 Frederik Tilmann

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License Version 2

Contributors of fixes and additions: V. Kuhlmann, J. Paisley, N. Becker, T. Doerges, K. Huebner Thanks to everyone who send in bug reports.

1 NAME

latexrevise - selectively remove markup and text from latexdiff output

2 SYNOPSIS

```
latexrevise [ OPTIONS ] [ diff.tex ] > revised.tex
```

3 DESCRIPTION

latexrevise reads a file `diff.tex` (output of *latexdiff*), and remove the markup commands. If no filename is given the input is read from standard input. The command can be used in *ACCEPT*, *DECLINE*, or *SIMPLIFY* mode, or can be used to remove user-defined latex commands from the input (see **-c**, **-e**, **-m**, and **-n** below). In *ACCEPT* mode, all appended text fragments (or preamble lines) are kept, and all discarded text fragments (or preamble lines) are deleted. In *DECLINE* mode, all discarded text fragments are kept, and all appended text fragments are deleted. If you wish to keep some changes, edit the `diff.tex` file in advance, and manually remove those tokens which would otherwise be deleted. Note that *latexrevise* only pays attention to the `\DIFaddbegin`, `\DIFaddend`, `\DIFdelbegin`, and `\DIFdelend` tokens and corresponding FL varieties. All `\DIFadd` and `\DIFdel` commands (but not their contents) are simply deleted. The commands added by *latexdiff* to the preamble are also removed. In *SIMPLIFY* mode, `\DIFaddbegin`, `\DIFaddend`, `\DIFdelbegin`, `\DIFdelend` tokens and their corresponding FL varieties are kept but all other markup (e.g. `\DIFadd` and `<\DIFdel>`) is removed. The result will not in general be valid latex-code but it will be easier to read and edit in preparation for a subsequent run in *ACCEPT* or *DECLINE* mode. In *SIMPLIFY* mode the preamble is left unmodified.

4 OPTIONS

-a or --accept

Run in *ACCEPT* mode (delete all blocks marked by `\DIFdelbegin` and `\DIFdelend`).

-d or --decline

Run in *DECLINE* mode (delete all blocks marked by `\DIFaddbegin` and `\DIFaddend`).

-s or --simplify

Run in *SIMPLIFY* mode (Keep all `\DIFaddbegin`, `\DIFaddend`, `\DIFdelbegin`, `\DIFdelend` tokens, but remove all other *latexdiff* markup from body).

Note that the three mode options are mutually exclusive. If no mode option is given, *latexrevise* simply removes user annotations and markup according to the following four options.

-c cmd or --comment=cmd

Remove `\cmd{...}` sequences. `cmd` is supposed to mark some explicit annotations which should be removed from the file before release.

-e envir or --comment-environment=envir

Remove explicit annotation environments from the text, i.e. remove

```
\begin{envir}
...
\end{envir}
```

blocks.

-m cmd or --markup=cmd

Remove the markup command `\cmd` but leave its argument, i.e. turn `\cmd{abc}` into `abc`.

-n envir or --markup-environment=envir

Similarly, remove `\begin{envir}` and `\end{envir}` commands but leave content of the environment in the text.

-V or --verbose

Verbose output

-q or --no-warnings

Do not warn users about `\DIDadd{..}` or `\DIFdel{..}` statements which should have been removed already.

5 BUGS

The current version is a beta version which has not yet been extensively tested, but worked fine locally. Please send bug reports to tilmann@esc.cam.ac.uk. Include the serial number of *latexrevise* (from comments at the top of the source). If you come across latexdiff output which is not processed correctly by *latexrevise* please include the problem file as well as the old and new files on which it is based, ideally edited to only contain the offending passage as long as that still reproduces the problem.

latexrevise gets confused by commented `\begin{document}` or `\end{document}` statements

6 SEE ALSO

latexdiff

7 PORTABILITY

latexrevise does not make use of external commands and thus should run on any platform supporting PERL v5 or higher.

8 AUTHOR

Copyright (C) 2004 Frederik Tilmann

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License Version 2

1 NAME

latexdiff-vc - wrapper script that calls latexdiff for different versions of a file under version management (CVS, RCS or SVN)

2 SYNOPSIS

```
latexdiff-vc [ latexdiff-options ] [ latexdiff-vc-options ] -r [rev1] [-r rev2] file1.tex  
[ file2.tex ...]
```

or

```
latexdiff-vc [ latexdiff-options ] [ latexdiff-vc-options ] [ --postscript | --pdf ]  
old.tex new.tex
```

3 DESCRIPTION

latexdiff-vc is a wrapper script that applies *latexdiff* to a file, or multiple files under version control (CVS or RCS), and optionally runs the sequence of `latex` and `dvips` or `pdflatex` commands necessary to produce pdf or postscript output of the difference tex file(s). It can also be applied to a pair of files to automatise the generation of difference file in postscript or pdf format.

4 OPTIONS

--rcs, --svn, or --cvs

Set the version system. If no version system is specified, latexdiff-vc will venture a guess.

latexdiff-cvs and latexdiff-rcs are variants of latexdiff-vc which default to the respective versioning system. However, this default can still be overridden using the options above.

-r, -r *rev* or --revision, --revision=*rev*

Choose revision (under RCS, CVS or SVN). One or two `-r` options can be specified, and the resulting in different behaviour:

latexdiff-vc -r *file.tex* ...

compares *file.tex* with the most recent version checked into RCS.

latexdiff-vc -r *rev1* *file.tex* ...

compares *file.tex* with revision *rev1*.

latexdiff-vc -r *rev1* -r *rev2* *file.tex* ...

compares revisions *rev1* and *rev2* of *file.tex*.

Multiple files can be specified for all of the above options. All files must have the extension `.tex`, though.

latexdiff-vc *old.tex new.tex*

compares two files.

The name of the difference file is generated automatically and reported to stdout.

-d or **--dir** **-d** *path* or **--dir=***path*

Rather than appending the string `diff` and optionally the version numbers given to the output-file, this will prepend a directory name `diff` to the original filename, creating the directory and subdirectories should they not exist already. This is particularly useful in order to clone a complete directory hierarchy. Optionally, a pathname *path* can be specified, which is prepended instead of `diff`.

--ps or **--postscript**

Generate postscript output from difference file. This will run the sequence `latex; latex; dvips` on the difference file (do not use this option in the rare cases, where three `latex` commands are required if you care about correct referencing). If the difference file contains a `\bibliography` tag, run the sequence `latex; bibtex; latex; latex; dvips`.

--pdf

Generate pdf output from difference file using `pdflatex`. This will run the sequence `pdflatex; pdflatex` on the difference file, or `pdflatex; bibtex; pdflatex; pdflatex` for files requiring `bibtex`.

--force

Overwrite existing diff files without asking for confirmation. Default behaviour is to ask for confirmation before overwriting an existing difference file.

--help or **-h**

Show help text

--version

Show version number

All other options are passed on to `latexdiff`.

5 SEE ALSO

latexdiff

6 PORTABILITY

latexdiff-vc uses external commands and is therefore limited to Unix-like systems. It also requires the RCS version control system and latex to be installed on the system. Modules from Perl 5.8 or higher are required.

7 AUTHOR

Copyright (C) 2005 Frederik Tilmann

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License Version 2 Contributors: S Utcke, H Bruyninckx

A simple example

We start with a draft text, `example-draft.tex`, listed here in full but also included in the distribution (except that the “`verbatim`” environment had to be renamed to “`Verbatim`” for the listing).

```
\documentclass[12pt,a4paper]{article}

\setlength{\topmargin}{-0.2in}
\setlength{\textheight}{9.5in}
\setlength{\oddsidemargin}{0.0in}

\setlength{\textwidth}{6.5in}

\title{latexdiff Example - Draft version}
\author{F Tilmann}

\begin{document}
\maketitle

\section*{Introduction}

This is an extremely simple document that showcases some of latexdiff features.
Type
\begin{Verbatim}
latexdiff -t UNDERLINE example-draft.tex example-rev.tex > example-diff.tex
\end{Verbatim}
to create the difference file. You can inspect this file directly. Then run either
\begin{Verbatim}
pdflatex example-diff.tex
xpdf example-diff.pdf
\end{Verbatim}
or
\begin{Verbatim}
latex example-diff.tex
dvips -o example-diff.ps example-diff.dvi
gv example-diff.ps
\end{Verbatim}
to display the markup.

\section*{Another section title}

A paragraph with a line only in the draft document. More things
could be said were it not for the constraints of time and space.

More things could be said were it not for the constraints of time and space.

And here is a typo.

Here is a table:

\begin{tabular}{ll}
Name & Description \\ \hline
Gandalf & Grey \\
Saruman & White
\end{tabular}

And sometimes a whole paragraph gets completely rewritten. In this
case latexdiff marks up the whole paragraph even if some words in it
are identical.
No change, no markup!
\end{document}
```

We can now edit this text as we would do with any other latex file to create a new revision of the text, `example-rev.tex`. We should run

latex example-rev.tex

and look at the resulting .dvi file to make sure that all changes are valid. An example revision is listed here:

```
\documentclass[12pt,a4paper]{article}

\setlength{\topmargin}{-0.2in}
\setlength{\textheight}{9.5in}
\setlength{\oddsidemargin}{0.0in}

\setlength{\textwidth}{6in}

\title{latexdiff Example - Revised version}
\author{F Tilmann}
% Note how in the preamble visual markup is never used (even
% if some preamble might eventually end up as visible text.)

\begin{document}
\maketitle

\section*{Introduction}

This is an extremely simple document that showcases some of the latexdiff features.
Type
\begin{Verbatim}
latexdiff -t UNDERLINE example-draft.tex example-rev.tex > example-diff.tex
\end{Verbatim}
to create the difference file. You can inspect this file directly. Then run either
\begin{Verbatim}
pdflatex example-diff.tex
xpdf example-diff.pdf
\end{Verbatim}
or
\begin{Verbatim}
latex example-diff.tex
dvips -o example-diff.ps example-diff.dvi
gv example-diff.ps
\end{Verbatim}
to display the markup.

\section*{Yet another section title}

More things could be said were it not for the constraints of time and space.

A paragraph with a line only in the revised document.
More things could be said were it not for the constraints of time and space.

And here is a typo.

Here is a table:

\begin{tabular}{ll}
Name & Description \\ \hline
Gandalf & White \\
Saruman & Evil
\end{tabular}

And now for something completely different, with not a paragraph in sight.
No change,
no markup!
\end{document}
```

To compare both revisions, type

```
latexdiff -t UNDERLINE example-draft.tex example-rev.tex > example-diff.tex
```

This results in the following difference file (a few newlines have been added in this listing for legibility reasons):

```

\documentclass[12pt,a4paper]{article}

\setlength{\topmargin}{-0.2in}
\setlength{\textheight}{9.5in}
\setlength{\oddsidemargin}{0.0in}

%DIF 7c7
%DIF < \setlength{\textwidth}{6.5in}
%DIF -----
\setlength{\textwidth}{6in} %DIF >
%DIF -----

%DIF 9c9
%DIF < \title{latexdiff Example - Draft version}
%DIF -----
\title{latexdiff Example - Revised version} %DIF >
%DIF -----
\author{F Tilmann}
% Note how in the preamble visual markup is never used (even %DIF >
% if some preamble might eventually end up as visible text.) %DIF >
%DIF PREAMBLE EXTENSION ADDED BY LATEXDIFF
%DIF UNDERLINE PREAMBLE %DIF PREAMBLE
\RequirePackage[normalem]{ulem} %DIF PREAMBLE
\RequirePackage{color} %DIF PREAMBLE
\providecommand{\DIFadd}[1]{\color{blue}\uline{#1}} %DIF PREAMBLE
\providecommand{\DIFdel}[1]{\color{red}\sout{#1}} %DIF PREAMBLE
%DIF SAFE PREAMBLE %DIF PREAMBLE
\providecommand{\DIFaddbegin}{} %DIF PREAMBLE
\providecommand{\DIFaddend}{} %DIF PREAMBLE
\providecommand{\DIFdelbegin}{} %DIF PREAMBLE
\providecommand{\DIFdelend}{} %DIF PREAMBLE
%DIF FLOATSAFE PREAMBLE %DIF PREAMBLE
\providecommand{\DIFaddFL}[1]{\DIFadd{#1}} %DIF PREAMBLE
\providecommand{\DIFdelFL}[1]{\DIFdel{#1}} %DIF PREAMBLE
\providecommand{\DIFaddbeginFL}{} %DIF PREAMBLE
\providecommand{\DIFaddendFL}{} %DIF PREAMBLE
\providecommand{\DIFdelbeginFL}{} %DIF PREAMBLE
\providecommand{\DIFdelendFL}{} %DIF PREAMBLE
%DIF END PREAMBLE EXTENSION ADDED BY LATEXDIFF

\begin{document}
\maketitle

\section*{Introduction}

This is an extremely simple document that showcases some of latexdiff features.
Type
\begin{Verbatim}
latexdiff -t UNDERLINE example-draft.tex example-rev.tex > example-diff.tex
\end{Verbatim}
to create the difference file. You can inspect this file directly. Then run either
\begin{Verbatim}
pdflatex example-diff.tex
xpdf example-diff.pdf
\end{Verbatim}
or
\begin{Verbatim}
latex example-diff.tex
dvips -o example-diff.ps example-diff.dvi
gv example-diff.ps
\end{Verbatim}
to display the markup.

\section*{\DIFaddbegin \DIFadd{Yet another } \DIFaddend \DIFdelbegin
\DIFdel{Another } \DIFdelend section title}

```

```

\DIFFdelbegin \DIFFdel{A paragraph with a line only in the draft
document. }\DIFFdelend More things could
be said were it not for the constraints of time and space.

\DIFFaddbegin \DIFFadd{A paragraph with a line only in the revised
document. }\DIFFaddend More things could be said
were it not for the constraints of time and space.

And here is a \DIFFaddbegin \DIFFadd{typo}\DIFFaddend \DIFFdelbegin
\DIFFdel{typo}\DIFFdelend .

```

Here is a table:

```

\begin{tabular}{ll}
Name & Description \\
\hline
Gandalf & \DIFFaddbegin \DIFFadd{White }\DIFFaddend \DIFFdelbegin
\DIFFdel{Grey }\DIFFdelend \\
Saruman & \DIFFaddbegin \DIFFadd{Evil
}\DIFFaddend \DIFFdelbegin \DIFFdel{White
}\DIFFdelend \end{tabular}

```

```

And \DIFFaddbegin \DIFFadd{now for something completely different, with not
a paragraph in sight}\DIFFaddend \DIFFdelbegin \DIFFdel{sometimes a whole
paragraph gets completely rewritten. In this
case latexdiff marks up the whole paragraph even if some words in it
are identical}\DIFFdelend .
No change,
no markup!
\end{document}

```

Type

```

pdflatex example-diff.tex
xpdf example-diff.pdf

```

to make the markup visible. This is what it looks like:

latexdiff Example - Revised version

F Tilmann

October 19, 2007

Introduction

This is an extremely simple document that showcases some of [the](#) latexdiff features.
Type

```
latexdiff -t UNDERLINE example-draft.tex example-rev.tex > example-diff.tex
```

to create the difference file. You can inspect this file directly. Then run either

```
pdflatex example-diff.tex  
xpdf example-diff.pdf
```

or

```
latex example-diff.tex  
dvips -o example-diff.ps example-diff.dvi  
gv example-diff.ps
```

to display the markup.

~~Another~~ [Yet another](#) section title

~~A paragraph with a line only in the draft document.~~ More things could be said were it not for the constraints of time and space.

[A paragraph with a line only in the revised document.](#) More things could be said were it not for the constraints of time and space.

And here is a ~~tip~~[typo](#).

Here is a table:

Name	Description
Gandalf	Grey White
Saruman	White Evil

And ~~sometimes a whole paragraph gets completely rewritten. In this case latexdiff marks up the whole paragraph even if some words in it are identical~~[now for something completely different. with not a paragraph in sight.](#) No change, no markup!

If you approve of all the changes in the revision, just continue with `example-rev.tex`

for the next revision. If you like to adopt most but not all changes you can use `latexrevise` in the following manner. Simply remove the `\DIFdelbegin` and `\DIFdelend` tags around the text you would like to keep and simply remove the text between `\DIFaddbegin` and `\DIFaddend` tags, if you do not wish to keep them. Say you are happy with all proposed changes for the example above except in the last paragraph where you prefer the original draft. You have to change

```
...
And \DIFaddbegin \DIFadd{now for something completely different, with not
  a paragraph in sight}\DIFaddend \DIFdelbegin \DIFdel{sometimes a whole
  paragraph gets completely rewritten. In this
case latexdiff marks up the whole paragraph even if some words in it
are identical}\DIFdelend .
...
```

into

```
...
And \DIFdel{sometimes a whole
  paragraph gets completely rewritten. In this
case latexdiff marks up the whole paragraph even if some words in it
are identical}.
...
```

and run

```
latexrevise -a example-rev.tex > example-final.tex
```

`example-final.tex` is then be almost identical to `example-rev.tex` except for the last paragraph.