

texchanges

LaTeX-native track changes

Phuc Nguyen

Version 0.2.4, 15 August 2026

Abstract

Texchanges records additions, removals, replacements, highlights, and comments directly in LaTeX source. The same document can show review markup, proposed final text, or the original text. Registered authors, stable change IDs, and decision statuses support human review and automated editing workflows.

Contents

1	Installation	1
2	Automatic latexdiff	2
3	Core commands	2
4	Authors, IDs, and decisions	2
5	Document modes	2
6	Reports	3
7	Presentation and localization	3
8	Compatibility with changes	3
9	Resolving source markup	3
10	Resources and license	3

1 Installation

Texchanges is distributed through TeX Live. Install it with the normal package manager when needed and load it with:

```
\usepackage[review]{texchanges}
```

For Overleaf projects whose TeX Live version does not contain Texchanges, upload `texchanges.sty` beside the main document. The unified `texchanges-overleaf.zip` bundle includes the package, the explicit review document, the two automatic comparison revisions, and a top-level `latexmkrc`. Select `texchanges-explicit-review.tex` as Main for explicit review, or `texchanges-review.tex` as Main for automatic diff.

2 Automatic latexdiff

The automatic example uses word-level matching. A removed word such as “new” is marked separately while a short shared phrase such as “system provides” remains unchanged. The example also demonstrates phrase replacement, a sentence-end insertion, a change inside `\emph{}`, and a revised `itemize` list. The conditional `latexmkrc` runs `latexdiff` only when `texchanges-review.tex` is selected as the Main document.

Use explicit Texchanges markup or manual review for changed display math, complex floats, and custom verbatim-like environments.

3 Core commands

The native API uses the source order “old, new” for replacements.

```
\txadd{new text}
\txremove{old text}
\txreplace{old text}{new text}
\txhighlight{important text}
\txcomment{Check this claim.}
```

The short aliases `\add`, `\remove`, and `\replace` are installed only when another package has not defined them.

4 Authors, IDs, and decisions

```
\txdefineauthor[name={Phuc Nguyen},color=orange]{phuc}

\txreplace[
  author=phuc,
  id=R12,
  comment={Clarify this claim.},
  status=pending
]{old text}{new text}
```

The optional keys are `author`, `id`, `comment`, and `status`. Status accepts `pending`, `accepted`, or `rejected`. Non-empty change IDs must be unique. A legacy optional label such as `\txadd[Reviewer]{text}` remains available for visual labels without structured author meta-data.

5 Document modes

Mode	Pending	Accepted	Rejected
review	marked change	proposed text	original text
final	proposed text	proposed text	original text
original	original text	original text	original text

Final mode warns when pending changes remain. Change lists and summaries are review artifacts, so `\txlistofchanges` produces no output in final or original mode.

6 Reports

Detailed reports include excerpts, IDs, page numbers, and links when `hyperref` is loaded. Summary reports group counts by author, type, and status. Run LaTeX twice after changes to refresh report data.

```
\txlistofchanges
\txlistofchanges[style=summary]
\txlistofchanges[
  title={Pending reviewer changes},
  show={added,replaced,commented},
  status={pending},
  author={phuc}
]
```

7 Presentation and localization

Presets are `texchanges`, `default`, `underlined`, `bfit`, and `nocolor`. Individual added, deleted, highlighted, comment, and author renderers can be selected through package options or replaced at runtime. English, British English, German, French, Italian, and Vietnamese report labels follow Babel or Polyglossia when their caption hooks are available.

8 Compatibility with changes

Compatibility is opt in because the replacement argument order differs.

```
\usepackage[review,compat=changes]{texchanges}
\definechangesauthor[name={Phuc Nguyen},color=orange]{phuc}
\replaced[id=phuc,changeid=R12]{new text}{old text}
```

The compatibility option provides documented `changes` commands and setter aliases. Its `id` key identifies an author, while `changeid` stores the Texchanges change ID. The `commandname` prefix option accepts `none`, `ifneeded`, or `always`.

9 Resolving source markup

The supplemental `texchanges-merge` tool can update decisions or remove resolved markup permanently from any working directory after TeX Live installs the executable.

```
texchanges-merge paper.tex reviewed.tex --accept
texchanges-merge paper.tex final.tex --accept --merge
texchanges-merge paper.tex --reject --id R12 --in-place
```

From a source checkout, run the script directly:

```
python3 scripts/texchanges-merge.py --version
```

The tool requires Python 3.10 or later and uses only the standard library.

In-place edits create a backup. The parser handles nested groups and skips comments and common verbatim-like environments. Malformed input fails before the destination is written.

10 Resources and license

Project documentation, examples, releases, and issue reporting are available at <https://phucnht.github.io/texchanges/> and <https://github.com/phucnht/texchanges>.

Texchanges is maintained by Phuc Nguyen. It may be distributed and modified under the LaTeX Project Public License, version 1.3c or later.