

# Git and latexdiff: compare versions of LaTeX documents

7 October 2015

Before using Git, I often used the [latexdiff](#) Perl script to highlight the differences between different versions of my LaTeX documents. For those of you not familiar with this tool, the following line in a terminal:

```
latexdiff version1.tex version2.tex > diff2.tex
```

will compare `version1.tex` to `version2.tex` and create a new file `diff2.tex` showing the differences.

## Example:

For a simple example, suppose version 1 of our LaTeX document is the following:

### 1 Showcasing latexdiff

latexdiff is a great tool that is very useful and works on the command line. Try it out!

Euler's identity (incorrect):

$$e^{2i\pi} + 3 = 0 \tag{1}$$

We notice a few errors, so we fix them in version 2 below:

### 1 Showcasing latexdiff

latexdiff is an awesome tool that is very useful. Try it out - it also works with Git!

Euler's identity:

$$e^{i\pi} + 1 = 0 \tag{1}$$

If we want to easily see the differences between the two versions, latexdiff will produce the following:

# 1 Showcasing latexdiff

latexdiff is ~~a great~~ an awesome tool that is very useful ~~and works on the~~ command line. Try it out - it also works with Git!

Euler's identity ~~(incorrect)~~: :-:

$$e^{\underline{2i\pi i\pi}} + \underline{31} = 0 \tag{1}$$

The tex and pdf files for these three versions are below:

- [version1.pdf](#) (created from [version1.tex](#))
- [version2.pdf](#) (created from [version2.tex](#))
- [diff.pdf](#) (created from [diff.tex](#))

Recently, while using Git for version control for some LaTeX files, I thought it would be great if one could somehow use latexdiff to see the differences between commits. A quick search on [Stackoverflow](#) revealed that [someone awesome](#) had already created a way to do this!

The tool is called [git-latexdiff](#), and instructions on how to install it and use it are below.

## Installation:

---

Before installing `git-latexdiff`, first check that `latexdiff` is properly installed on your system, by entering the following in a terminal:

```
latexdiff -V
```

If you see something like:

```
This is LATEXDIFF 1.1.1 (Algorithm::Diff 1.15 so, Perl v5.18.2)
```

then it is installed correctly and you can move on to installing `git-latexdiff`.

If you do not see this, or see a strange error message (**Oct 2016**: which might happen in the latest version of Ubuntu), then check out [this post](#).

## Installation of git-latexdiff:

---

### Step 1: Create a temporary folder

In a terminal, create a temporary folder to which you can download the git-latexdiff installation files, and then move there e.g.

```
mkdir -p ~/build
cd ~/build
```

(The `-p` is very useful - it allows one to create intermediate directories, and will not return an error if the folder already exists.)

## Step 2: clone the git-latexdiff repo

In your temp folder, download the installation files using:

```
git clone https://gitlab.com/git-latexdiff/git-latexdiff.git
```

This will create a `~/build/git-latexdiff` subfolder.

## Step 3: Move to the subfolder

In the terminal:

```
cd ~/build/git-latexdiff
```

## Step 4: Run the installation script

In the terminal:

```
sudo make install
```

(you will probably need to enter your password)

## Step 5: Add the folder to your PATH

In your `~/.bashrc` (Linux) or `~/.bash_profile` (OSX) file, add the following:

```
# gitlatex path:
export PATH=~/build:$PATH
```

This allows the terminal to find the `git latexdiff` command (below).

Installation over!

## How to use git-latexdiff:

---

Suppose you are working on a LaTeX document `main.tex`, and that you have created several commits using Git.

To see the difference between the current (HEAD) version of `main.tex` and the previous commit's version, in a terminal (in the folder containing `main.tex`) simply type:

```
git latexdiff HEAD~1 --main main.tex
```

and a PDF showing the changes (like [diff.pdf](#)) should pop up.

### Note 1:

If you have only one tex file in the folder, you do not need use the `--main` flag, and can simply use:

```
git latexdiff HEAD~1
```

In fact, even if you have multiple tex files (perhaps you are using `\input{}` or `\include{}` in your `main.tex`), `git-latexdiff` is quite smart and will probably guess the main file correctly.

### Note 2:

Suppose you want to compare the current commit to the version two commits back. You then use:

```
git latexdiff HEAD~2
```

rather than `HEAD~1`. If you want to compare to the N commits back, then use `HEAD~N`:

```
git latexdiff HEAD~N
```

Finally, if you want to compare an older commit to a previous commit, then first do a checkout before using `git-latexdiff`.

### Note 3:

If your git repo has different folders, e.g. your

- `main.tex` is in `/myproject/tex/`, but
- `references.bib` is in `/myproject/formatting/`

then you need to run `latexdiff` in `myproject` using the `--main` flag and a path to your `main.tex`:

```
cd ../myproject
git latexdiff HEAD~1 --main ./tex/main.tex
```

#### Note 4:

The Perl script latexdiff does not indicate differences between figures in LaTeX versions. Only the figures that are in the latest commit will be in the diff.pdf.

#### Note 5:

If you play around with the `~/.gitconfig` file and git-latexdiff stops working, simply re-install.

#### Note 6:

In case you want access to the generated diff.pdf files, on OSX they are stored in the `/private/tmp/` folder in subfolders `git-latexdiff.12345/` (with different random numbers instead of `12345`).

## Credits

---

Credit goes to [Matthieu Moy](#) for creating `git-latexdiff`. The following [post](#) pointed me to this solution (which in my opinion is the best of the posted solutions).

According to [this page](#) and [this page](#), the main author of latexdiff is [Frederik Tilmann](#).

Both these guys are absolutely awesome.