

# gitlog.sty

A proof of concept for automatic typesetting  
of change logs from the *git* DVCS

Brent Longborough

November 2015

Release: o.o.beta (3b2c518)

## CONTENTS

---

<b>Introduction</b>	<b>1</b>
Limitations . . . . .	1
How <i>gitlog</i> works . . . . .	1
<b>Using the package</b>	<b>2</b>
Package options . . . . .	2
General options . . . . .	2
Options for controlling the .bib file . . . . .	2
Options for linking to on-line services . . . . .	3
Typesetting the change log . . . . .	3
<b>Etc</b>	<b>4</b>
Release notes . . . . .	4
Acknowledgements . . . . .	4
Copyright & licence . . . . .	4
From the author . . . . .	4
<b>Sample Git Change Log</b>	<b>5</b>

## INTRODUCTION

---

The *git* distributed version control system maintains an historical log of update activity. The *gitlog* package provides a way automatically to typeset such a log, optionally linking commits from the typeset log to one of the online DVCS hosting services.

## Limitations

The current release (o.o.beta) is intended only as a *proof-of-concept*, and should not be used for production-level work unless you're happy with these limitations:

- Formatting maliciously-coded documents *can cause arbitrary files to be overwritten*. Although this problem is easy to avoid (and requires the *-shell-escape* command line option), there are no built-in protections.
- The *git* change log is built and formatted using the facilities of *biblatex* and *biber*. The way this is currently implemented makes it unlikely that documents using *gitlog* can contain 'normal' bibliographies.
- New lines in the *git* change log commit messages are simply converted to spaces. The result is pretty ugly, unless you've had the foresight to punctuate your commit messages nicely.

That said, I think *gitlog* will still be useful to a subset of T<sub>E</sub>X users, and I welcome suggestions and contributed code via email or Github, where the url is <https://github.com/Hightor/gitlog/issues>.

## How *gitlog* works

1. The package uses a *git* command internally, via the `\write18` command, to write out the change log in the shape of a rather strange *.bib* file
2. The change log *.bib* file contains a series of items of type *gitcommit*, defined and formatted by the *gitlog* data model and bibliography styles which accompany the package.
3. The rest of the work is done by persuading *biblatex* and *biber* to treat this data as a bibliography.
4. Optionally, you can generate your own 'change log bibliography', using this command (joined up into a single line):

```
git --no-pager log --reverse --date=short
--pretty="format:@gitcommit{%h,%n author = {%an},%n
date = {%ad}, %n title = {%B}, %n commithash = {%H}
}" > filename
```

## USING THE PACKAGE

---

To collect and typeset *git* history, you load the *gitlog* package in the usual way:

```
\usepackage[< options >]{gitlog}
```

## Package options

The following options are available:

### General options

#### **title=log title**

This option allows you to change the chapter title associated with the typeset change log. The default is ‘Change Log’.

#### **date, nodate**

These complementary options allow you to specify whether or not you want the author date and name to be added to the change log. The default is `nodate`.

### Options for controlling the .bib file

#### **write, nowrite**

These complementary options allow you to specify whether or not you want *gitlog* to generate the special change log (*.bib*) for you.

If you specify `write`, then *gitlog* will regenerate the change log every time the document is formatted. Note that this option is implemented with the `\write18` command, and requires that your document be processed with the `TeX` option `--shell-escape`.

If you specify `nowrite`, then *gitlog* will not write anything, and the `--shell-escape` option is not required. However, in this case, you are responsible for generating the change log in the correct format for *gitlog* to use, as well as being able to hand-tailor it.

If neither option is specified, the default depends on whether or not you use the `bibfile` option, described next.

**bibfile=filename**

The *git* change log data is kept in a file in *biblatex* *.bib* format, which *gitlog* writes (if requested) and then reads to format the change log.

If this option is not specified, then a default filename is used: `<job-name>.gitlog.bib`. In this case, the default option *write* is used, but can be suppressed by specifying *nowrite*.

This can be overridden by specifying your own choice of filename using this option. In this case, the default option *nowrite* is used, but you can force *gitlog* to write to your file by specifying *write*.

Warning: there is *no* protection against writing to any file name whatsoever. The default settings are reasonably ‘safe’; where you need to ship the document without its repository, then

`[write,bibfile=gitlog.local.bib]`

is probably a safe set of options to use.

**Options for linking to on-line services**

The change log typeset by *gitlog* can include links connecting each commit to its corresponding page in either the GitHub or the Atlassian Bitbucket online repository services. (Trade marks of their respective owners.)

To use this feature, you must load the *hyperref* package before loading *gitlog*, and use one of the following, mutually exclusive, options.

**github=repository-path**

When each commit in the changelog is typeset, a link is generated to the corresponding page on GitHub, with a URL in this format:

`https://github.com/repository-path/commit/commit-hash`

**bitbucket=repository-path**

When each commit in the changelog is typeset, a link is generated to the corresponding page on Atlassian Bitbucket, with a URL in this format:

`https://bitbucket.org/repository-path/commits/commit-hash`

**Typesetting the change log**

The command `\printGitLog` typesets the change log, as a new chapter containing a pseudo-bibliography, at that point in the document. At the moment, there are very few options to allow you to tailor the typography.

A sample change log can be seen at the end of this manual.

ETC

---

## Release notes

Ro.o.beta: 2015-11-24 Initial beta release

## Acknowledgements

The T<sub>E</sub>X.SE community has been a constant source of help, inspiration, and amazement.

I'd also like to register my thanks to the owners of the packages and tools on which *gitlog* depends: *biblatex*, *etoolbox*, *hyperref*, and *kvoptions*, and also to moewe at T<sub>E</sub>X.SE for invaluable help during my ongoing stumblings with *biblatex*.

We all depend heavily on the constant hard work so unstintingly given to the community by the Guardians of CTAN. Thank you all.

The failings, of course, I claim for myself.

## Copyright & licence

Copyright © 2015, Brent Longborough, who has asserted his moral right to be identified as the author of this work.

This work — *gitlog* — may be distributed and/or modified under the conditions of the LaTeX Project Public License: either version 1.3 of this license, or (at your option) any later version.

The latest version of this license can be found at the L<sup>A</sup>T<sub>E</sub>X Project website,<sup>1</sup> and version 1.3 or later is part of all recent distributions of L<sup>A</sup>T<sub>E</sub>X.

This work has the LPPL maintenance status 'maintained'; the Current Maintainer of this work is Brent Longborough.

## From the author

Although my limitations as a T<sub>E</sub>Xnician mean that I'm going to need a lot of expert help to turn *gitlog* into a delightful author experience, I hope you find the package useful and the concept interesting. I'll be very happy to receive your comments by email.

Brent Longborough

brent+ctancontrib (bei) longborough (punkt) org  
and at T<sub>E</sub>X.SE

---

<sup>1</sup>(<http://www.latex-project.org/lppl.txt>)

## SAMPLE GIT CHANGE LOG

---

- [9dce109] *Initial commit.*
- [f558cdd] *First code to build pseudo-bibfile.*
- [240f4c9] *Added data model (not working) - now we step back a bit.*
- [b4629f6] *First working version full of restrictions.*
- [7f69aba] *Merge branch 'feature/Basic.Function' into develop.*
- [d651d24] *Gradually improving things (especially understanding).*
- [4c1dd2c] *Making an MWE for TeX.SX.*
- [9ed8b6f] *commithash display solved.*
- [601242b] *Linking the change log to GitHub!*
- [981d564] *Merge branch 'feature/Basic.Function' into develop.*
- [65c44ee] *Make title a verbatim field Now ready for first bash at packaging and documentation.*
- [201cdf1] *Merge branch 'feature/Basic.Function' into develop.*
- [11ca996] *Initial packaging complete.*
- [b65bf5a] *Fix for git R.2 problem.*
- [63ba36f] *Tagged with gitinfo2.*
- [9cde412] *Added makefile, new documentation file.*
- [afcea87] *Improve handling of gitlog bib file Tailor filename and suffix Improved defaults.*
- [4562bf2] *Change gitlog bib file defaults Assume nowrite for non-default bib file name.*
- [d798eba] *Merge branch 'feature/Refine.Options' into develop.*
- [7ca98b4] *Tidy the (test) documents; remove local dependency.*
- [fa237e9] *Ensure auxfiles cleaned up before build.*
- [b4b7a78] *First published results, as a pdf.*
- [8aaa18d] *Implemented commit link option for Bitbucket.*
- [e009031] *Filling out the README.*
- [a9eb541] *Slight (PEBKAC) struggle with Markdown.*
- [3410e54] *More Markdown lessons needed.*
- [03ae070] *Starting in on the documentation Refining Makefile.*
- [f1d08b0] *Generate sample gitlog.bib outside of TeX (no write18) Ensure clean build.*
- [83d3553] *Working on manual.*

- [89599b6] *Finalise documentation for first release.*
- [965b231] *Merge branch 'feature/Early.Documentation' into develop.*
- [546f296] *Prepare for Release o.o.beta.*
- [34e186e] *Merge branch 'release/o.o.beta' into CTAN.*
- [3b2c518] *Ensure correct file permissions (again!)*