

Git-ify Your (digital) Life

Git-based tools to ease your life

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de> | JSC Internal Seminar

Overview

`Git` *a short review*

`etckeeper` *keep your system's configs*

`vcsh` *version your \$HOME*

`mr` *my / multiple repositories*

`git-annex` *so meta!*

`bup` *backup with Git*

`ikiwiki` *Wiki compiler and publisher in a Git repo*

`gcrypt` *PGP-encrypted Git repositories*

`gitodo` *cmd-based ToDo List Manager in a Git repo*

`Tipps` *I can't resist ...*

Git-ify Your (digital) Life

Part I: Git

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

Version Control System *Git*

A short overview

- decentralized / distributed
alike Mercurial/hg or Bazaar in contrast to CVS or Subversion
- designed by Linus “*the-one-and-only*” Torvalds
Git is British-English slang for “*unpleasant person*”
- non-linear history
branching and *merging* is easy and performant
- cryptological verification of revisions
each revision (*commit*) has a unique SHA-1 hash computed from diff + meta info
- no need for a server / everything is locally available
because of first point

Git-ify Your (digital) Life

Part II: etckeeper

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

etckeeper

Keep Your System's Configurations¹

- creates a Git (or Mercurial/Bazaar/Darcs) repo for `/etc`
- uses additional meta-file for remembering permissions for each file
DVCS usually don't track file owner info; only executable bit
- uses *pre-commit* hooks to fix file permissions
- hooks itself into package managers (e.g. *apt*, *zypper*) to auto-commit `/etc` before and after package changes
- manual commits also possible

¹ Windows users: sleep or think of moving to Linux

Git-ify Your (digital) Life

Part III: vcsh

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

vcsh – Version Control System for (your) \$HOME

version `.profile`, `.{bash,zsh,vim}rc`, ... without pollution

- separate Git repositories for dotfiles
without polluting \$HOME with `.git` directories
- easily migrate your personalized environment to other hosts
clone your `.vim` repository on new host to have it synchronized
- allows for different branches for different hosts
e.g. “`tklatt-zamws`”, “`myself-laptop`”, “`su-myserver`”
- *vcsh* is a single Shell script

vcsh – Version Control System for (your) \$HOME

Example

One Repository for Your Vim Config²

```
vcsh init vim
vcsh vim add ~/.vimrc ~/.vim
vcsh vim commit -m 'Initial commit of my Vim configuration'

# optionally push your files to a remote
vcsh vim remote add origin <remote>
vcsh vim push -u origin master

# from now on you can push additional commits like this
vcsh vim push
```

²taken from vcsh's Readme

Git-ify Your (digital) Life

Part IV: mr

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

mr – my / multiple repositories

Git-ify Your (digital) Life

Part V: git-annex

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

git-annex – Version Files Without Their Contents

So meta! ³

- saves meta info (i.e. name, size) of files without their contents
- saves actual files read-only in `.git/annex/objects`
symlinks them to original/real location
- keeps track of which remote has which files
each remote identified by UUID
- designed for flaky connections
uses rsync for data transfer

³ Windows users: Wake up!

git-annex – Version Files Without Their Contents

I mean, so really meta!

- written in Haskell
- allows for special remotes
 - Amazon S3 / Glacier
 - WebDAV
 - rsync
 - the web (`http(s)://`, `ftp://`, `archive.org`, `arxiv.org/[format]/[ID]`, etc.)
 - podcast feeds
 - XMPP
 - simple directories
 - gcrypt
- example collection of conference proceedings (slides + video recordings)
`https://github.com/RichiH/conference_proceedings`

git-annex – Version Files Without Their Contents

Example Szenario: The Archivist

- annex all files
- actual files offline in special remotes on USB drives, tapes, etc.
- having full information about name, size and location of all files in one place at hand

Example

```
#> git annex whereis
whereis my_cool_big_file (1 copy)
    7570b02e-15e9-11e0-adf0-9f3f94cb2eaa  -- backup drive
whereis other_file (3 copies)
    0c443de8-e644-11df-acbf-f7cd7ca6210d  -- here (laptop)
    62b39bbe-4149-11e0-af01-bb89245a1e61  -- usb drive
    7570b02e-15e9-11e0-adf0-9f3f94cb2eaa  -- backup drive
```

git-annex – Version Files Without Their Contents

Example Szenario: The Nomad

- keep copies of data online (on internet)
- sync several local devices for occasional backup
- add data locally while on the road
- sync data to online remotes while at Internet café or friend's place
- perfect for photos while traveling

Git-ify Your (digital) Life

Part VI: bup

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

bup – Git for LARGE Files

- recap: Git is designed for plaintext files
binary files are just a huge blob for Git; no diff possible
- uses Git object trees and replaces hashing and packing algorithms
- designed for space-saving incremental backups
- backups can be FUSE mounted
- can be a special remote for git-annex
- written in Python

Git-ify Your (digital) Life

Part VII: ikiwiki

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

ikiwiki – Wiki Compiler and Publisher in a Git Repo

■

Git-ify Your (digital) Life

Part VIII: gcrypt

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

gcrypt – PGP-encrypted Git repositories

- implements a git-remote-handler to deal with `gcrypt : remotes`
- can be a special remote for git-annex
- Remark: You might want to use Joey “joeyh” Hess’ fork of gcrypt

Git-ify Your (digital) Life

Part IX: gitodo

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

gitodo – ToDo List Manager with Git

- simple commandline-based ToDo list management
- Todos are managed in a single Git repository
- single portable Ruby script (was a Shell script some time ago)

Example

```
# gitodo
S  Pri      Deadline      ID      Subject
---+-----+-----+-----+-----+-----+-----+-----+-----+
  | -3 |                | 2 | Christmas presents
D | 0 |                | 9 | Remove garbage from your bed
C | 0 |                | 8 | Go to sleep -- yes, it's important
  | 0 | 2013-12-10 23:59 | 10 | Still awake?
  | 0 | 2013-12-24      | 7 | Remove SVN from all computers
```


Git-ify Your (digital) Life

Part X: Tipps

January 15, 2014 | Torbjörn Klatt <t.klatt@fz-juelich.de>

Tipps

I can't resist ...

- portable GUI for Git (browsing and actions): *git-cola*⁴
- Zsh
 - very powerful built-in completion for most programs (Git: incl. selecting branches/tags)
 - prompt-integrated info about current VCS working copy
 - highly customizable prompt (left and right)
 - can mimic Bash, Ksh, tcsh (never tried it myself)
- Finally:

You cannot time travel in real-life.
But you can in your digital life, selectively!

⁴<https://github.com/git-cola/git-cola>

Project Links

etckeeper <https://joeyh.name/code/etckeeper/>

vcsh <https://github.com/RichiH/vcsh>

mr <https://github.com/joeyh/myrepos>

git-annex <https://git-annex.branchable.com/>

bup <https://github.com/bup/bup>

ikiwiki <https://ikiwiki.info/>

gcrypt <https://github.com/blake2-ppc/git-remote-gcrypt>

gitodo <https://github.com/vain/gitodo>

Sources

- This talk is heavily inspired by Richard “RichiH” Hartman’s talk at *Linuxtag 2013*⁵
- official and unofficial documentation of named tools
- (long-term) experiments with named tools

⁵ <http://www.linuxtag.org/2013/de/program/mittwoch-22-mai-2013.html?eventid=147>

Thank you for your interest!

Questions?

(now or later)