

Advanced Git Commands You Will Actually Use



stosb.com/talks

Tom Hacoen
Samsung Open Source Group

tom.hacoen@samsung.com
[@TomHacoen](https://twitter.com/TomHacoen)

Quick Setup Before We Begin

- In case you do not have these set already:

```
$ git config --global user.name "Tom Hacoheh"  
$ git config --global user.email "tom.hacoheh@samsung.com"  
$ git config --global color.ui true  
$ export EDITOR="vim" # Optional  
$ git config --global diff.tool "vimdiff" # Extra optional
```

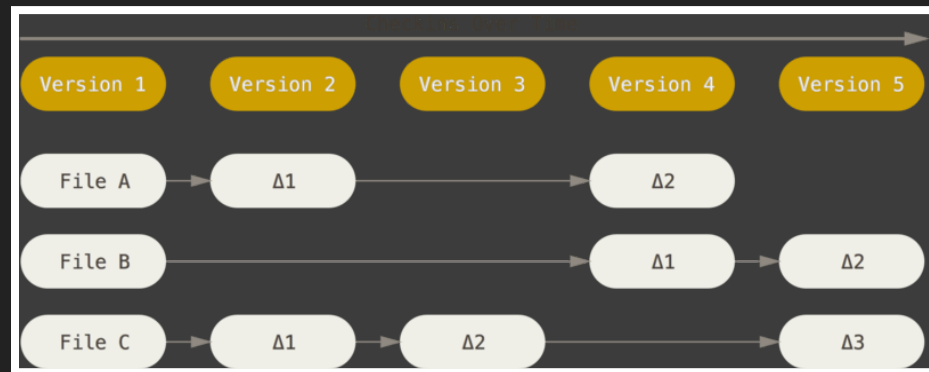
- Enable Git command autocompletion (system dependent)
- Set up aliases (here for reference)

```
$ git config --global alias.unstage 'reset HEAD --'
```

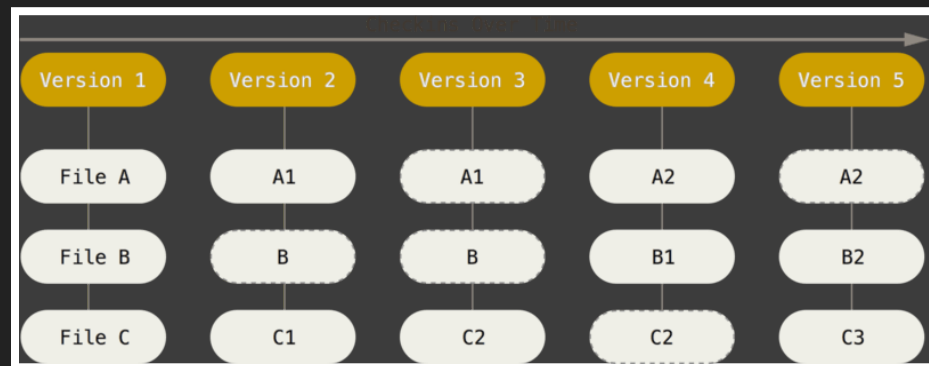
A Few Useful Properties of Git

Snapshots, Not Differences

- Differences

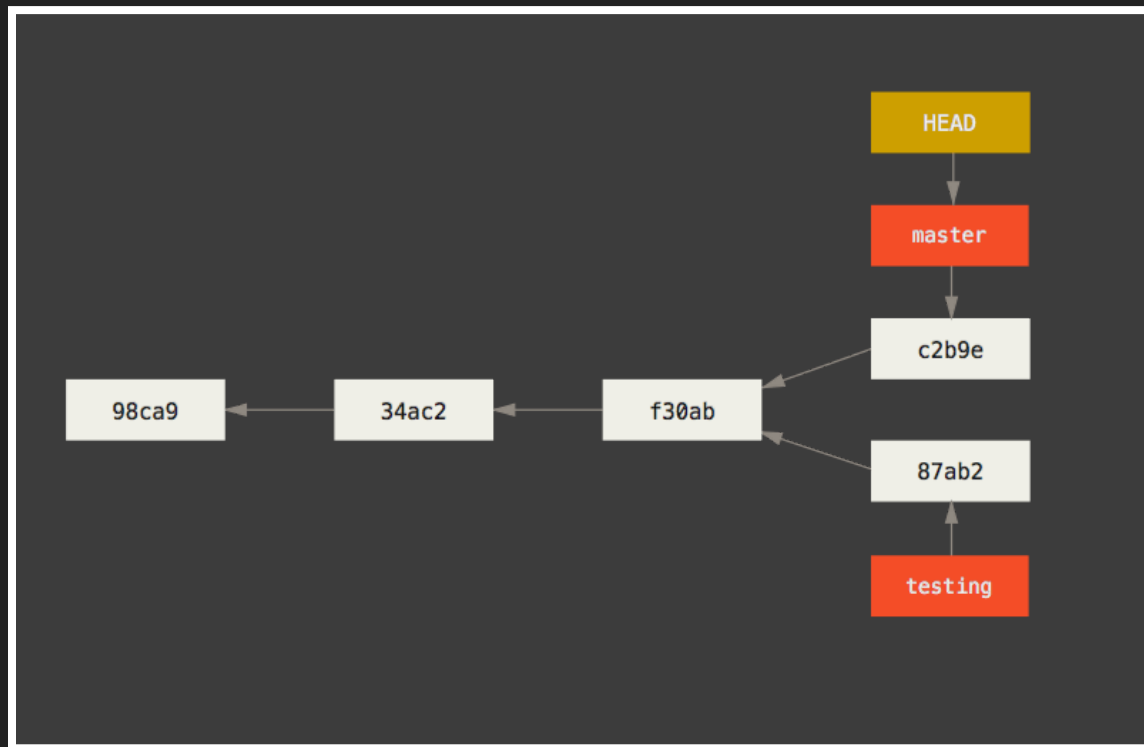


- Snapshots



A Few Useful Properties of Git

Branches and Tags are References



A Few Useful Properties of Git

Nearly Every Operation Is Local

- Operations are fast
- You can work offline
- You can be sloppy (and fix later)

A Few Useful Properties of Git

Git Generally Only Adds Data

- Easy to see older (even temporary) versions
- Almost everything is reversible
- Safe to experiment with

A Few Useful Properties of Git

Just a Few More...

- Has a staging area
- History is mutable (easy to abuse)
- Everything is checksummed

Reading the Log

Cleaner Tree Log Overview

```
$ git log --oneline --graph
* 7b3fa8a Migrate elementary to the new Eo4 syntax
|
| \
| * 02e87e8 Fix warnings following migration to Eo4.
| * 7bd1d48 Map: Correct broken migration.
| * e74ec8c Automatic migration to Eo4.
| * 9274efb Remove redundant defines.
|
| ✓
* 36669f1 Scaling test: reorder instructions to set the correct
scale
* b9c912f radio: inherit from elm check
```


Reading the Log

Listing a Change Summary

```
$ git log --stat
commit a31f399857ecf9409e6aa6fb8effe9477ee47fe2
Author: Tom Hacoen <tom@stosb.com>
Date:   Wed Jun 22 16:52:05 2016 +0100

    Edje object: Add API for replacing the internal text object

src/lib/edje/edje_load.c      | 34 ++++++-----
src/lib/edje/edje_object.eo | 17 ++++++++
src/lib/edje/edje_private.h  |  1 +
src/lib/edje/edje_util.c     | 32 ++++++++
4 files changed, 60 insertions(+), 24 deletions(-)
```

Reading the Log

Seeing Actual Code Changes

```
$ git log -p
commit c66d478ebb2c1a0aff52571467d7a6b87a533382
Author: Ji-Youn Park <jy0703.park@samsung.com>
Date: Thu Mar 24 17:54:05 2016 +0830

    Elm_image: remove Elm_Image_Orient.

diff --git a/src/lib/elm_image.c b/src/lib/elm_image.c
***** Snip Snip *****
}

EOLIAN static void
-_elm_image_orient_set(Eo *obj, Elm_Image_Data *sd, Elm_Image_Orient
orient)
+_elm_image_efl_image_orientation_set(Eo *obj, Elm_Image_Data *sd,
Efl_Gfx_Orientation orient)
{
    if (sd->edje) return;
```

Reading the Log

Limiting Commits

```
# Commits changing file/function in file
$ git log main.c
$ git log -L :list_find:main.c

# Commits containing string
$ git log --grep FOOBAR # Messages containing string
$ git log -S FOOBAR # Lines containing string

# Commits in HEAD, not in master
$ git log master..

# Commits that are in either "foo" or "bar" (not both)
$ git log foo...bar

# Only show the first parent ("main" history)
$ git log --first-parent
```

Reading the Log

Finding Out What You Have Been Up To

```
# The total number of commits by an author
$ git shortlog -nse --author=tom.hacohen@samsung.com

# The total number of commits by an author in the last year
$ git shortlog -nse --author=tom --since="1 year ago"

# A list of commits by an author in the last year
$ git log --author=tom --since="1 year ago"
```

Reading the Log

Getting a Version Description

```
# Get a standard version description (requires at least one tag)
$ git describe --long
v1.17.0-236-gc66d478

# Get the SVN-like monotonic revision number
$ git rev-list --count HEAD
12514
```

Inspecting Commits and States

Seeing Into the Past

```
# Showing the changes in a commit
$ git show 80f14e8fea0057ee950f0778dd51b096ca9850a4
$ git show my_branch # Can be branch, tag or whatever.

# Showing a file from a different state
$ git show v1.7.0:main.c

# Switching working directory to a different reference
$ git checkout c9b306777 # Or any other reference
```

Branches

Viewing

```
# All the branches (including remote)
$ git branch -a
# Use "git fetch -p" to clean up stale remote branches

# All branches that are fully contained in HEAD
$ git branch -a --merged

# All branches that are not full contained in HEAD
$ git branch -a --no-merged
```

Branches

Manipulating

```
# Rebase branch over the upstream version
$ git pull --rebase # Can be set in config

# Rebase branch over a specific branch
$ git rebase origin/master

# Merge a branch and always create a merge commit
$ git merge --no-ff

# Rebase and keep the branch structure
$ git pull --rebase=preserve
$ git rebase --preserve-merges origin/master

# Applying a commit from a different branch
$ git cherry-pick 80f122437d
```


Making Changes

Inspecting Workspace State

```
# A more condensed status
$ git status -s

# Changes compare to upstream
$ git diff origin/master

# Seeing the diff of the staging area
$ git diff --cached

# Ignore whitespace changes in diff
$ git diff -w
```

Making Changes

Adding Files to the Staging Area

```
# Adding parts of a file  
$ git add -p file # File can also be a dir, or omitted  
  
# Adding all of the changed files in a directory  
$ git add -u src/
```

Making Changes

Using the Stash

```
# Stashing all of the changes
$ git stash

# Stashing some of the changes
$ git stash -p

# Applying back the stash
$ git stash apply

# Stash has many more features I do not use
$ git stash --help
```

Rewriting History

Un-staging Files

```
$ git status -s
M README

$ git reset README
# git reset # for all the files
$ git status -s
M README

$ git checkout README
# git checkout -f # for all the files
$ git status -s
# Nothing
```

Rewriting History

Editing the Most Recent Commits

```
# Remove the most recent commits and their changes
$ git reset --hard HEAD^
$ git reset --hard HEAD~3 # Or any other pointer (for a range)
$ git reset --hard origin/master # Reset the state to upstream

# Keep the changes uncommitted
$ git reset HEAD^
$ git reset c9b306777 # Or any other pointer

# Merging index into the most recent commit
$ git add NEWS
$ git commit --amend # Also lets you edit the commit message
# Add -v to git commit to also see the diff

# Edit the author
$ git commit --author "007 <jb@mi6.gov.uk>" --amend
```

Rewriting History

The Most Useful Command in The World

```
$ git rebase -i HEAD~5
pick 7b07b03 track/manage size hints for zoomap child objects
pick 71a85b7 update winlist ui when using directional selection
pick bbd4d2f force changed when adding keyboards
pick a424542 disable emotion_shutdown during shutdown procedure

# Commands:
# p, pick = use commit
# r, reword = use commit, but edit the commit message
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
# f, fixup = like "squash", but discard this commit's log message
# x, exec = run command (the rest of the line) using shell
# d, drop = remove commit
#
# These lines can be re-ordered; executed from top to bottom.
# If you remove a line here THAT COMMIT WILL BE LOST.
```

Rewriting History

Recovering Lost Commits

```
# Jump to a hash state if you know it
$ git checkout c9b306777

# Find unreferenced (missing) commits
$ git reflog
cebf78d HEAD@{0}: rebase -i (finish): returning to refs/heads/master
cebf78d HEAD@{1}: rebase -i (start): checkout HEAD^
c6e355f HEAD@{2}: rebase finished: returning to refs/heads/master
c6e355f HEAD@{3}: pull --rebase: Elementary test entry: Create an
editable test object.
83b0592 HEAD@{4}: commit: Elementary test entry: Create an editable
test object.
2fd8861 HEAD@{5}: commit (amend): Ui text: Add an editable variant
(tiny wrapper).
```

Rewriting History

Removing Parts of a Commit

```
# Commit c42bc3a535 (can be anywhere in history)
$ git revert -n c42bc3a535
$ git reset # Remove everything from staging

# Add back the wanted changes
$ git add NEWS # All of this file
$ git add -p # Some parts of the rest

# Merge the commit into the original commit
# Either amend if it is the HEAD
$ git commit --amend
$ git checkout -f # Remove the rest of the changes
# Or fixup if anywhere else
$ git commit -m "Temp"
$ git checkout -f # Remove the rest of the changes
$ git rebase -i c42bc3a535^ # Mind the ^ (caret)
```


Delivering Changes

Getting Your Commits Out There

```
# Change the url of the repository
$ git remote set-url origin ssh://git@newserver.com/repo.git

# Adding a new remote
$ git remote add new ssh://git@alt.newserver.com/repo.git

# Using the new remote
$ git fetch new
$ git rebase new/master
$ git push new master

# Generate patch files for a series of commits
$ git format-patch HEAD~5 # Or any other reference
```

Investigating Bugs

Finding Who Added a Line and Why

```
# Check who changed the file
$ git blame eo.c # Add "-w" to ignore white-space

06f65ab2 eo.c (Tom Hacohe 2016-05-19 11:33:17 +0100
    321)          vtable = &klass->vtable;
fc880379 eo.c (Tom Hacohe 2015-11-09 11:45:04 +0000
    322)          inputklass = main_klass = klass;
7be0748b eo.c (J r my Zurcher 2013-07-30 15:02:35 +0200
    323)
c2b4137f eo.c (Carsten Haitzler 2015-10-24 12:23:53 +0900
    324)          if (!cache->op)

# At an earlier revision
$ git blame fc880379^ -- eo.c
```

Investigating Bugs

Finding When a Bug Was Introduced

```
$ git bisect start
# To limit bisect to a directory: "git bisect start -- src/"
$ git bisect bad COMMIT
$ git bisect good COMMIT
Bisecting: 417 revisions left to test after this (roughly 9 steps)
[7352bcff98fc65a08edcd505b872403af8d821a7] edge_external: fix
external icon handling
$ git bisect good
Bisecting: 208 revisions left to test after this (roughly 8 steps)
[9f5d27972252d67fe92ca44a1c610da4ed531b86] Evas events: Implement
support for hold event

# ... SNIP ...

a31f399857ecf9409e6aa6fb8effe9477ee47fe2 is the first bad commit
```

Investigating Bugs

Automatic Bisect

```
$ cat ~/test.sh
#!/bin/sh
make || exit 125           # this skips broken builds
~/check_issue.sh         # does the test case pass?

$ git bisect start HEAD HEAD~10 -- # Last 10 commits
$ git bisect run ~/test.sh
$ git bisect reset         # quit the bisect session

a31f399857ecf9409e6aa6fb8effe9477ee47fe2 is the first bad commit
```

A Useful Tool - tig

Main View

```
2016-06-06 15:22 Daniel Kolesa
2016-06-06 15:18 Stefan Schmidt
2016-06-06 15:16 Stefan Schmidt
2016-06-06 14:49 Stefan Schmidt
2016-06-06 14:14 Andy Williams
2016-06-06 00:01 Andy Williams
2016-06-05 11:12 Tom Hacothen
2016-06-03 14:17 Cedric Bail
2016-06-03 14:04 Cedric Bail
2016-06-03 13:55 Cedric Bail
2016-06-02 17:19 Cedric Bail
2016-05-27 16:48 Tom Hacothen
2016-05-19 22:17 Andy Williams
2016-05-18 23:36 Andy Williams
2016-05-17 23:28 Andy Williams
2016-05-17 17:25 Andy Williams
2016-05-16 23:15 Jean-Philippe ANDRE
2016-05-16 11:59 Andy Williams
2016-05-16 19:52 Jean-Philippe ANDRE
2016-05-16 19:33 Jean-Philippe ANDRE
2016-05-16 19:28 Jean-Philippe ANDRE
2016-05-16 11:01 Andy Williams
2016-05-16 18:03 Jean-Philippe ANDRE
2016-05-16 01:24 Jean-Philippe ANDRE
2016-05-14 11:20 Andy Williams
2016-04-28 22:51 Andy Williams
2016-04-28 22:32 Andy Williams
2016-04-24 22:48 Andy Williams
2016-04-14 10:48 Hermet Park
2016-04-09 09:48 Hermet Park
2016-04-07 17:44 Andy Williams
2016-04-07 17:43 Andy Williams
2016-04-07 16:27 Andy Williams
2016-03-31 15:42 Andy Williams
2016-03-30 16:44 Andy Williams
2016-03-30 03:58 Andy Williams
2016-03-22 13:10 Andy Williams
2016-03-18 19:42 Andy Williams
2016-03-22 13:10 Andy Williams
2016-03-18 19:42 Andy Williams
2016-03-10 00:06 Andy Williams
2016-03-08 22:53 Andy Williams
2016-03-01 21:20 Andy Williams
2016-02-25 15:51 Andy Williams
2016-02-25 15:50 Andy Williams
2016-02-21 23:46 Andy Williams
2016-02-21 18:48 Andy Williams
2016-02-21 18:39 Andy Williams
[main] 8809ed16298e0a72e869ba781b34fe10766eae6b - commit 730 of 49139

o eolian: add a new references system to help
o examples: eio: remove unused variable
o examples: eio: mark unused parameter as such
o tests: evas: remove unused variable
o [devs/tasn/elm_code] elementary_code: Fix t
o elementary_code: Use correct theme file pos

Merge elm code, a code editing widget into
o (origin/devs/cedric/elm_code) elementary_
o elementary: make elm_code a BETA API.
o elementary: integrate elm_code theme.
o elementary: merging in elm_code widget.
Merge in elm code widget
o [elm_code] elm_code: Fix build for upda
o elm_code: Fix undo code for deleting ta
o elm_code: Revert an optimisation to fix
o elm_code: Fix crash when tabbing in a s
o elm_code: unbreak previous commit
o elm_code: ensure lines are visible when
o elm_code: fix crash from previous commi
o elm_code: remove useless variable
o elm_code: Fix scroller size when append
o [elm_code] don't double free standard p
o elm_code: Fix log test case
o tests: add a test case for log viewer
o Fix build for latest API
o [editor] fix issue where selections did
o [editor] Stop jumping around on click
o Update to latest eo specs
o fix build break
o remove duplicated variable declaration.
o [help] Stub out a right-click-help UI.
o [selection] split words on * and & also
o [editor] fix line geometry when scroll
o [editor] Split out lines from a stand
o [editor] move tooltips to inline erro
o [editor] split out the grid into rend
o Fix the local lookup of diffs in the
o Update to latest eo_add syntax
o Fix the local lookup of diffs in the
o Update to latest eo_add syntax
o Update to latest eo_add syntax
o Update to latest eo syntax
o Fix make dist
o Update to new eo event callback signatu
o [editor] draw line width marker after f
o [editor] a bunch of optimisations for r
o [editor] add delete and backspace undo
o Complete the ending newline crash fix
o Fix crash when undoing a soft tab inser

commit 8809ed16298e0a72e869ba781b34fe10766eae6b
Refs: v1.17.0-15288-g8809ed1
Merge: 14c8ddc 4120714
Author: Tom Hacothen <tom@stosb.com>
AuthorDate: Sun Jun 5 11:12:13 2016 +0100
Commit: Tom Hacothen <tom@stosb.com>
CommitDate: Mon Jun 6 14:28:16 2016 +0100

Merge elm code, a code editing widget into the efl

It was decided to merge this and mark the code as BETA for now.
It's currently being used by EDI and should also be used by inventor and etc.

data/elementary/themes/Makefile.am | 1 +
data/elementary/themes/default.edc | 1 +
data/elementary/themes/edc/elm_code.edc | 16 ++
src/Makefile_Elementary.am | 64 ++++
src/bin/elementary/gltionore | 1 +
src/bin/elementary/elm_code_test_main.c | 439 ++++++
src/bin/elementary/elm_code_test_private.h | 6 +
src/lib/elementary/Elementary.h | 1 +
src/lib/elementary/elm_code.c | 115 ++++++
src/lib/elementary/elm_code.h | 115 ++++++
src/lib/elementary/elm_code_common.h | 87 ++++++
src/lib/elementary/elm_code_diff_widget.c | 131 ++++++
src/lib/elementary/elm_code_diff_widget.h | 37 +++
src/lib/elementary/elm_code_file.c | 321 ++++++
src/lib/elementary/elm_code_file.h | 89 ++++++
src/lib/elementary/elm_code_line.c | 234 ++++++
src/lib/elementary/elm_code_line.h | 117 ++++++
src/lib/elementary/elm_code_parse.c | 206 ++++++
src/lib/elementary/elm_code_parse.h | 41 +++
src/lib/elementary/elm_code_private.h | 39 +++
src/lib/elementary/elm_code_text.c | 327 ++++++
src/lib/elementary/elm_code_text.h | 73 ++++++
src/lib/elementary/elm_code_widget.c | 1858 ++++++
src/lib/elementary/elm_code_widget_eo | 266 ++++++
src/lib/elementary/elm_code_widget_legacy.h | 13 +
src/lib/elementary/elm_code_widget_private.h | 66 ++++++
src/lib/elementary/elm_code_widget_selection.c | 454 ++++++
src/lib/elementary/elm_code_widget_selection.h | 44 +++
src/lib/elementary/elm_code_widget_text.c | 197 ++++++
src/lib/elementary/elm_code_widget_undo.c | 69 ++++++
src/tests/elementary/elm_code_file_test_load.c | 162 ++++++
src/tests/elementary/elm_code_file_test_memory.c | 49 +++
src/tests/elementary/elm_code_test_basic.c | 28 ++
src/tests/elementary/elm_code_test_line.c | 83 ++++++
src/tests/elementary/elm_code_test_parse.c | 102 ++++++
[diff] 8809ed16298e0a72e869ba781b34fe10766eae6b - line 1 of 62 774
```

A Useful Tool - tig

Blame

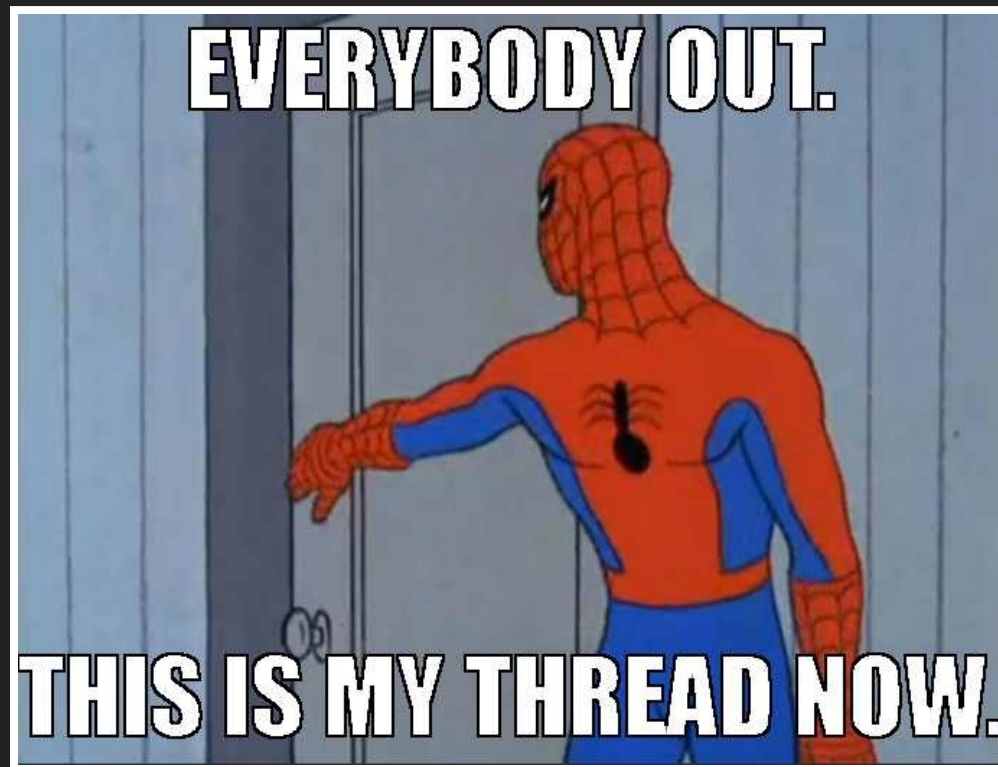
```
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e --(obj->user_refcount);
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e if (EINA_UNLIKELY(obj->user_refcount <= 0))
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e {
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e     if (obj->user_refcount < 0)
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e     {
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e         ERR("Obj:%p. User refcount (%d) < 0. Too many unrefs.", obj, obj->user_refcount);
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e         return;
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e     }
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e     _eo_unref(obj);
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e     }
2016-07-12 10:26 Tom Hacohe... legacy/eobj/lib/ee.c a267989 }
2016-07-12 10:26 Tom Hacohe... legacy/eobj/lib/ee.c a267989 }
2012-05-06 12:03 Tom Hacohe... legacy/eobj/lib/eeobj.c a41960d ERPI int
2012-04-11 07:05 Tom Hacohe... src/lib/ee/ee.c 337fac0 eo_ref_get(const Eo *obj_id)
2012-04-11 07:05 Tom Hacohe... legacy/eobj/lib/eeobj.c a41960d {
2012-04-10 14:19 Daniel Zaui... src/lib/ee/ee.c 337fac0     EO_OBJ_POINTER_RETURN_VAL(obj_id, obj, 0);
2012-04-23 08:10 Tom Hacohe... legacy/eobj/lib/eeobj.c 0075c23
2016-07-12 10:26 Tom Hacohe... src/lib/ee/ee.c a5eb66e     return obj->user_refcount;
2012-04-11 07:05 Tom Hacohe... legacy/eobj/lib/eeobj.c a41960d }
2012-04-11 07:05 Tom Hacohe... legacy/eobj/lib/eeobj.c a41960d }
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab ERPI void
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab eo_del_intercept_set(Eo *obj_id, Eo_Del_Intercept del_intercept_func)
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab {
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab     EO_OBJ_POINTER_RETURN(obj_id, obj);
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab     obj->del_intercept = del_intercept_func;
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab }
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab }
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab ERPI Eo_Del_Intercept
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab eo_del_intercept_get(const Eo *obj_id)
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab {
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab     EO_OBJ_POINTER_RETURN_VAL(obj_id, obj, NULL);
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab     return obj->del_intercept;
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab }
2016-03-08 16:57 Carsten Haitzler... src/lib/ee/ee.c 3df71ab }
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 9c929ba void
2013-04-18 14:19 Daniel Zaui... src/lib/ee/ee.c 337fac0 _eo_condtor_done(Eo *obj_id)
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 9c929ba {
2013-04-18 14:19 Daniel Zaui... src/lib/ee/ee.c 337fac0     EO_OBJ_POINTER_RETURN(obj_id, obj);
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 402bad0     if (obj->condtor_done)
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 402bad0     {
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 402bad0         ERR("Object %p is already constructed at this point.", obj);
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 402bad0         return;
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 402bad0     }
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 402bad0 }
2012-06-10 14:04 Tom Hacohe... legacy/eobj/lib/ee.c 9c929ba     obj->condtor_done = EINA_TRUE;
[blame] src/lib/ee/ee.c - line 1441 of 1816
```

Name Dropping

A Few Commands You Should Read More About

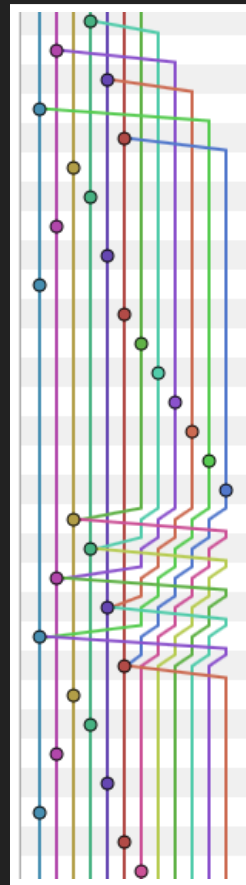
- `git send-email`
- `git checkout -b`
- `git-svn`, `git-hg...`
- `gitg`
- `gitk`

HIJACKING THIS TALK



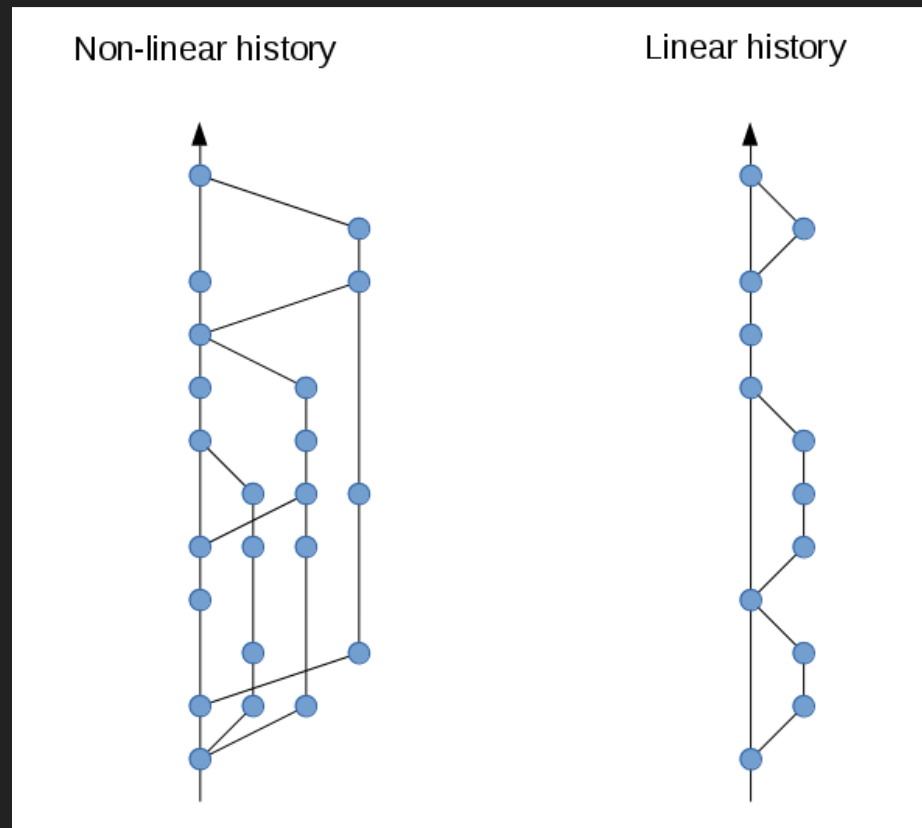
A Better Workflow

“I messed up Git so bad it turned into Guitar Hero”



A Better Workflow

Non-Linear vs. Linear



A Better Workflows

Maintaining a Linear History

```
# Maybe alias some of these
# Sync with upstream
$ git pull --rebase=preserve

# Merging a branch into master (working on branch feature)
$ git rebase --preserve-merges master
$ git checkout master
$ git merge --no-ff feature # Remove --no-ff for no merge commit
```

Commit Messages

Style Guideline

- Wrap message around 76 characters
- First line: brief, one-liner
- Second line should be left empty
- The body of the commit should be descriptive and verbose
- Use the imperative form for consistency with git merge and revert ("Fix bug" not "Fixed bug")
- Bonus: annotate with @fix, @feature, #NNN and etc.

Commit Messages

Example

```
Eo refcount: Split the refcount to private and public (user).
```

```
This commit changes the way refcount is dealt with internally. Before this commit, there was one refcount shared between Eo internals and users. Now there is a refcount for eo operations (like for example, function calls) and one for user refcount (eo_ref).
```

```
An example bug that this protects against (which is seemingly rather common) is:
```

```
some_eo_func(obj);
```

```
// Inside the implementation of that func:
```

```
pd->a = 1; // The object's private data
```

```
eo_unref(obj); // To delete the object
```

```
eo_unref(obj); // A big one extra unref
```

```
pd->a = 2; // Segfault, this data has already been freed
```

```
T3428
```

```
@feature
```

Questions?



stosb.com/talks

Tom Hacoen
Samsung Open Source Group

tom.hacoen@samsung.com
[@TomHacoen](https://twitter.com/TomHacoen)

Attribution

- The [Git Book](#)
- [Workflows Image #1](#)
- [Workflows Image #2](#)
- [Hijacking this talk](#)