

Git tips from the trenches

Day to day stuff I do with git

About me

- Staff Software Engineer at RubiconMD (now part of CVS / CVS Health)
- Training Engineers since 2005

Built a git course in 2010. Been teaching git since then.

🔗 <https://www.cursodegit.com>

✉️ cursos@cursodegit.com

💡 Need to level up your team? or you have specific git challenges? Reach out!

Table of contents

■ Life is too short to waste it...

1. ...typing
2. ...figuring out a git graph
3. ...wait for your AI agents
4. ...reviewing poorly crafted pull requests
5. ...using (only) git

Disclaimer

Opinions are my own and do not represent good practices or official guidelines.

This is how I work.

Use it if it works for you. Gracely ignore me if it doesn't.

Life is too short to waste it typing



Git aliases

Instead of writing this:

```
> git log --oneline --graph --decorate --all
```

write

```
> gloga
```

Chaining aliases

You can turn this...

```
> git switch main && git pull && git merge chemicalX && git  
push
```

...into

```
> gswm && gl && gm chemicalX && gp
```

Or this rebase operation...

```
> git switch chemicalX && git fetch --prune --all && git  
rebase origin/main && git push --force-with-lease
```

... into

```
> gsw chemicalX && gfa && grb origin/main && gp!
```



If you are lazy as I am...

... you don't have to type the aliases yourself!

- fish:
 - fish abbreviations (using fisher) <https://github.com/jhillyerd/plugin-git>
- zsh:
 - ohmyzsh git plugin <https://github.com/ohmyzsh/ohmyzsh/blob/master/plugins/git/README.md>
- bash:
 - ohmybash git plugin <https://github.com/ohmybash/oh-my-bash/tree/master/plugins/git>

Using git config

Using `git config` you can define aliases:

```
> git config --global alias.swm "switch main"
```

Then use them like this:

```
> git swm
```

Pros:

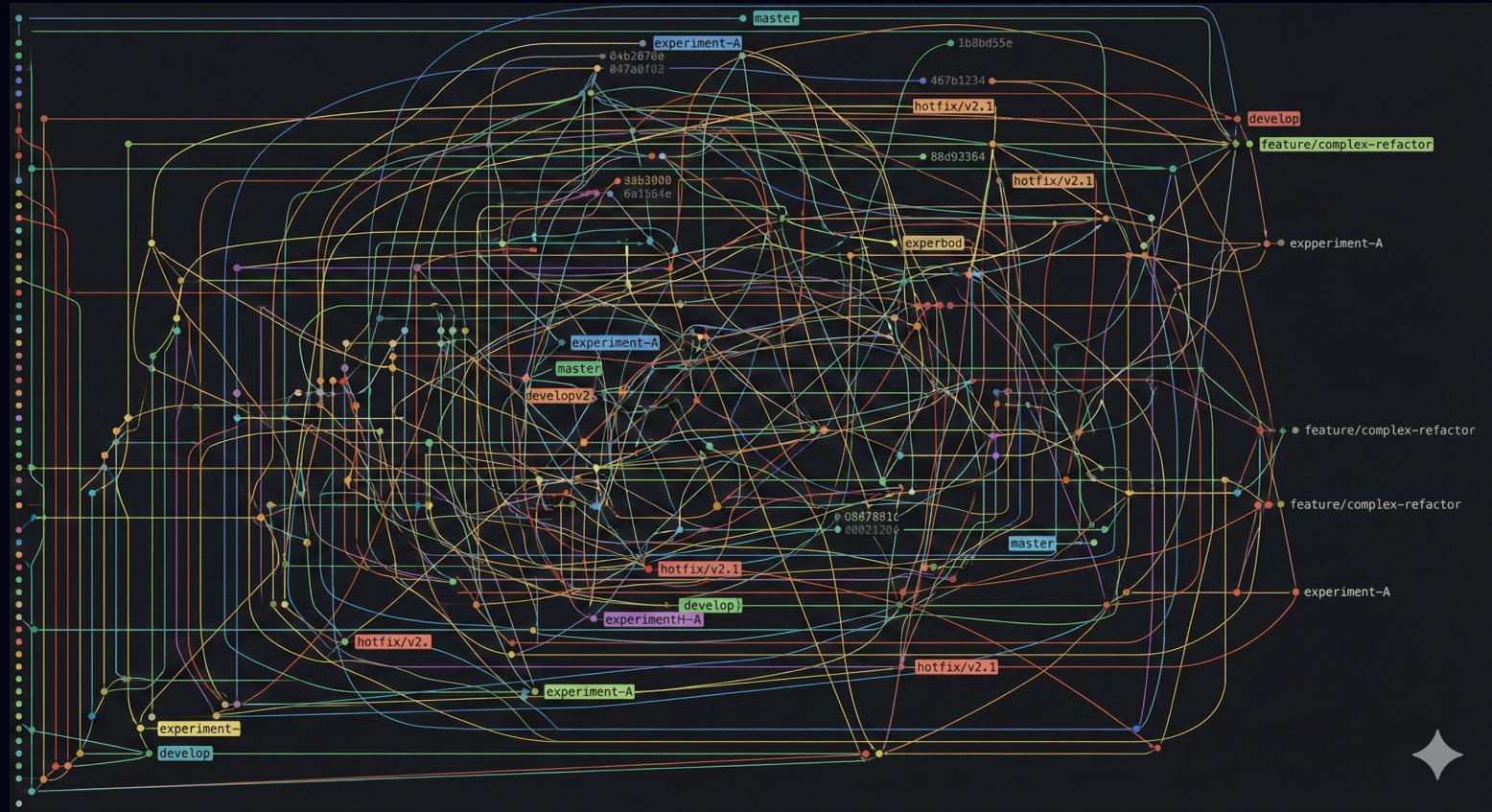
- use git configuration, valid in any shell

Cons:

- You still need to type `git` every time... and I'm too lazy for that

```
> git sw chemicalX && git fa && git rb origin/main && git swm && git l
```

Life is too short to waste it figuring out a git graph



Do I need to see the whole graph?

- Most of the time I just want to see what happened in a single branch. Or maybe two or three branches
- I don't care about all the intermediate commits, I don't want to see the trees in the forest



git log demo!!

```
vlc --fullscreen --mouse-hide-timeout 0 --no-video-title-show --play-and-exit git-tips-first-parent.mp4
```

snippet +exec is disabled, run with -x to enable

raw `git log` command

- `git log` provides too much information
- No graph structure
- Difficult to read
- Good when we need to look at the tree inside the forest

```
commit 7d6b970cbd3c216353cb22b383b70c150140662e (HEAD → master, origin/master, origin/HEAD)
Merge: 5357e18 960ad0
Author: Rajeev Karuvath <rajeevravindran@users.noreply.github.com>
Date:   Wed Sep 17 13:31:20 2025 -0700

    Merge pull request #264 from dboyd13/feat/amazon-bedrock-secrets

        feat: Amazon Bedrock API key patterns

commit 960adf0677665f7f7724e132013393fe956ae193
Author: Darran Boyd <dboyd@example.com>
Date:   Fri Aug 15 13:25:36 2025 +1000

        feat: Amazon Bedrock API key patterns

commit 5357e18bc27b42a827b6780564ea873a72ca1f01
Merge: 0ab55f8 53893ca
Author: Clarence "Sparr" Risher <clrnnc@amazon.com>
Date:   Tue Jun 20 19:28:33 2023 -0400

        Merge pull request #213 from jsoref/spelling

        spelling: appropriate

commit 0ab55f8446d49af9a62698bed748c53ea3fe789bb
Merge: bifdf0e0 b9e40d8
Author: Rogan Creswick <creswick@gmail.com>
Date:   Tue Jun 20 16:04:05 2023 -0700

        Merge pull request #221 from dannysauer/bug/no-require-terminal

        Only emit terminal codes to terminal (and don't require `say`)

commit b9e400833b028f9bde5451c760aa9fe713872de
Author: Clarence "Sparr" Risher <clrnnc@amazon.com>
Date:   Tue Jun 20 21:42:52 2023 +0000

        Replace which with command in tput check

        Signed-off-by: Clarence "Sparr" Risher <clrnnc@amazon.com>

commit bifdf0e0a52b7cb0af857c14a2673d0a2b7e365
Merge: 99d01d5 9538187
Author: Rogan Creswick <creswick@gmail.com>
Date:   Tue Jun 20 14:18:44 2023 -0700

        Merge pull request #224 from dannysauer/ci/test_pr

        run tests when PR opened
```

Using `git log` to display the full graph

- `git log --oneline --graph` allows us to see the graph structure
- Easier to read
- It still shows too much information:
 - What did happen in `main`?
 - Do I want to see all those intermediate commits?

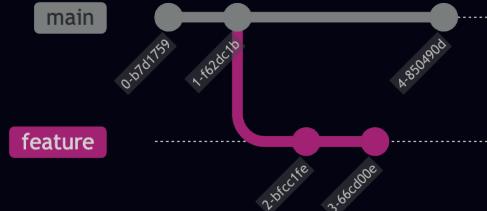
```
* 7d6b97b (HEAD → master, origin/master, origin/HEAD) Merge pull request #264 from dboyd13/feat/amazon-bedrock-secrets
|\ 
| * 960adfb feat: Amazon Bedrock API key patterns
|/|
| * 5357e18 Merge pull request #213 from jsoref/spelling
|/|
| * 53893ca spelling: appropriate
| | 9ab55f8 Merge pull request #221 from dannysauer/bug/no-require-terminal
|/|
| * | b9e40d8 Replace which with command in tput check
| | | 65891e2 Only emit terminal codes to terminal
| | | bif0fe0 Merge pull request #224 from dannysauer/ci/test_pr
|/|
|/|
| * | 9538187 run tests when PR opened
|/|
| * 99d01d5 Merge pull request #216 from sparr/master_branch
|/|
|/|
| * 0652a4d Explicitly set initial branch for tests
|/|
| * b9e96b3 Merge pull request #175 from dariuswiles/patch-1
|/|
| * 73sec09 Merge pull request #1 from dariuswiles/patch-2
|/|
| * 88ab351 Typo - "filter our false positives"
|/|
|/|
| * 1992161 Typo - "filter our false positives"
|/|
| * 80230af Merge pull request #173 from PatMyron/patch-2
|/|
| * be8854c fixing README bold formatting
|/|
| * 052b4cd Merge pull request #172 from PatMyron/patch-1
|/|
| * 23870b8 bold warning in README
|/|
| * 5e28af3 Merge pull request #116 from ScaleSec/master
|/|
| * b4a3208 Adding support for pre-commit framework
|/|
| * 3958dac Merge pull request #106 from fmariier/patch-1
|/|
| * ffe5378 Fix heading to resolve rst2man warning
|/|
| * ad82d68 (tag: 1.3.0) Prepare for 1.3.0 release
| | fd7da79 Merge pull request #103 from fmariier/manpage-whatis
|/|
| * baa4311 Fix generation of 'whatis' metadata from the manpage
| | db32f2a Merge pull request #101 from fmariier/fix-manpage-typo
|/|
| * | 3ba3fc9 Fix typo in manpage and README ("inadvertantly")
| | |
| * | e9b71e6 Merge pull request #100 from fmariier/issue99
|/|
|/|
```

--first-parent to the rescue!!

- `git log --oneline --graph --first-parent main` allows us to see what happened in the `main` branch
- Easier to read
- Displays what was merged into `main`
- We can see what was released in between two releases

```
* 7d6b970 (HEAD → master, origin/master, origin/HEAD) Merge pull request #264 from dboyd13/feat/amazon-bedrock-secrets
* 5357e18 Merge pull request #213 from jsoref/spelling
* 0ab55f8 Merge pull request #221 from dannysauer/bug/no-require-terminal
* b1fdfe0 Merge pull request #224 from dannysauer/ci/test_pr
* 99d01d5 Merge pull request #216 from sparr/master_branch
* b9e96b3 Merge pull request #175 from dariuswiles/patch-1
* 80230af Merge pull request #173 from PatMyron/patch-2
* 052b4cd Merge pull request #172 from PatMyron/patch-1
* 5e28df3 Merge pull request #116 from ScaleSec/master
* 3958dac Merge pull request #106 from fmarier/patch-1
* ad82d68 (tag: 1.3.0) Prepare for 1.3.0 release
* f7da7e9 Merge pull request #103 from fmarier/manpage-whatis
* db32f2a Merge pull request #101 from fmarier/fix-manpage-typo
* e9b71e6 Merge pull request #100 from fmarier/issue99
* c11c229 Merge pull request #90 from daneah/master
* 8450ad7 Merge pull request #89 from dbraley/master
* d8da475 Merge pull request #72 from dhoer/adv-config
* 2333a55 Merge pull request #67 from t13a/feature/fix-scan-files
* 1b5125d Merge pull request #82 from enabokov/exit-code-register-aws
* e324284 Merge pull request #88 from costleya/master
* b296988 Merge pull request #74 from hyandell/master
* 99800b9 Merge pull request #54 from teo-scout24/master
* c97cc2c Merge pull request #59 from mbacchi/stderr
* b10e08e Trim \n before testing if pattern is empty
* 74fc862 Updated git-secrets script to handle carriage returns and new lines when loading patterns
* aa3fb20 Merge pull request #40 from kramos/master
* 0714f2b Merge pull request #36 from joaufi/readme-fix
* 9f772c5 Merge pull request #35 from awslabs/fix-empty-providers
* d23f160 Fixing newline
* 0ecbdb2 Merge pull request #33 from initbar/master
* 3f1138c (tag: 1.2.1) 1.2.1 release
* 8619877 Explicitly set `IFS` when calling provider commands
* ae78ffd (tag: 1.2.0) 1.2.0 release
* 861c05e Merge pull request #25 from michaelwittig/fix_emptylines
* f0f4253 Merge pull request #28 from chrisgilmerproj/fix_recursive_scan
* 513af41 Merge pull request #26 from michaelwittig/fixr
```

life is too short to waste it figuring out a git graph



To merge `feature` into `main`:

```
> gswm  
> gm feature
```

This operation creates a `merge commit`



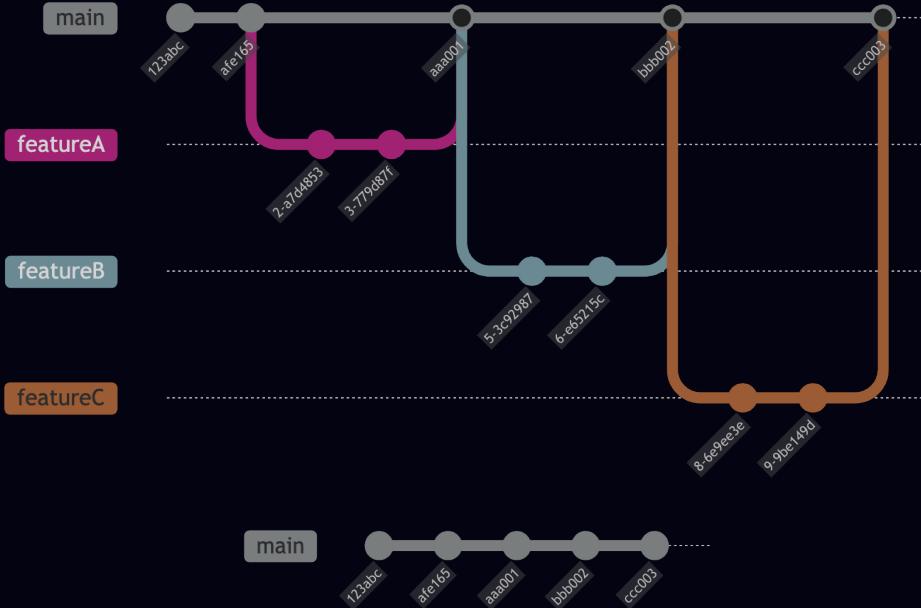
Anatomy of a merge commit

A `merge commit` has two parents:

- First parent is the commit at which HEAD (`main`) was pointing when the merge happened
- Second parent is the commit at which the branch we pull into HEAD (`dboyd13/feat/amazon-bedrock-secrets`) was pointing when the merge happened

```
1 > gsh
2 commit 7d6b970cbd3c216353cb22b383b70c150140662e (HEAD -> master, origin/master, origin/HEAD)
3 Merge: 5357e18 960adf0
4 Author: Rajeev Karuvath <rajeevravindran@users.noreply.github.com>
5 Date:   Wed Sep 17 13:31:20 2025 -0700
6
7     Merge pull request #264 from dboyd13/feat/amazon-bedrock-secrets
8
9     feat: Amazon Bedrock API key patterns
```

Life is too short to waste it figuring out a git graph



Life is too short to waste it waiting for your AI agents



Switching contexts: the two scenarios and how my former self addressed them

- Scenario 1: doing stuff that takes a long time to run (CI, builds, AI agents...)

Clone the repo in multiple directories

- Scenario 2: Work in multiple branches

Switch branches + stashing/unstashing changes

My actual self uses worktrees!

Worktrees allow you to checkout multiple branches at the same time in different directories

How this helps me?

- Switching contexts is instant (no need to stash/unstash changes)
- Can work on something else while waiting for other stuff (CI, AI agents, long builds...)

How do I switch contexts (worktrees)?

`tmux` to the rescue! 🧑

- Each project has its own session
- Use a different window for each worktree
- Create custom aliases to create and destroy sessions quickly
- Store my tmux sessions with `tmux-resurrect` plugin



Demo time!!

```
vlc --fullscreen --mouse-hide-timeout 0 --no-video-title-show --play-and-exit demo-worktrees-tmux.mp4
```

```
snippet +exec is disabled, run with -x to enable
```

⚠️ Caveats ⚠️

- Database configuration per worktree
- Environment variables per worktree
- Project setup (initial migrations, seeds, database dump restores...)

Remember why we do code reviews (sometimes we forget)

- catch bugs early
- share knowledge
- catch things you missed
- different (sometimes better) ways to do the same thing

The one rule I follow when crafting a PR

▀ ↳ Tell a story with my commits

This is the problem we want to fix...

...these are the steps that I took to fix it...

...this is why I fixed it this way

The one other rule is avoid AI slop in commit messages and PR descriptions

(two of) The tools that help me tell my story

■ Interactive staging (`git add -p`)

Several changes in my staging area, but I want to commit them separately

■ The fixup feature of `git`

Several changes in my staging area, but I want to add them to previous commits

Example: Fixup the commits in a repository

The branch was ready to push...

```
aalba@alfonsos-air ~/M/a/git-tips-fixup-demo (main)> git log --oneline --decorate --color --graph --all
* 1d35b0a (HEAD → main) Posts controller
* 3bb59b4 Post model
* 631fbc4 Migration and schema
* f034981 Scaffold
aalba@alfonsos-air ~/M/a/git-tips-fixup-demo (main)>
```

... then I realised I missed a field in the model

```
aalba@alfonsos-air ~/M/a/git-tips-fixup-demo (main)> git status
On branch main
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   app/controllers/posts_controller.rb
    modified:   app/models/post.rb
    modified:   app/views/posts/_form.html.erb
    modified:   app/views/posts/_post.html.erb
    modified:   app/views/posts/_post.json.jbuilder
    modified:   db/migrate/20260125101115_create_posts.rb
    modified:   db/schema.rb
    modified:   test/controllers/posts_controller_test.rb
    modified:   test/fixtures/posts.yml

no changes added to commit (use "git add" and/or "git commit -a")
aalba@alfonsos-air ~/M/a/git-tips-fixup-demo (main)>
```

I want the changes added to the right commits in the history



Fixup demo!

```
vlc --fullscreen --mouse-hide-timeout 0 --no-video-title-show --play-and-exit fixup-demo.mp4
```

snippet +exec is disabled, run with -x to enable

Fixup: two step process

1. Create the fixup commits
2. Rebase and squash

Life is too short even for so many git commands...

■ Welcome lazygit!

A terminal UI for git commands

<https://github.com/jesseduffield/lazygit>

```
vlc --fullscreen --mouse-hide-timeout 0 --no-video-title-show --play-and-exit lazygit-demo.mp4
```

snippet +exec is disabled, run with -x to enable

Life is too short to waste it using (only) git

▀ Say hello to other third generation VCS tools! jujutsu darcs/pijul sapling piper

- I've been using jujutsu for a couple of months already
- It has improved my git experience a lot

Feature	Git	Jujutsu (jj)
Primary Workflow	Explicit snapshots (<code>add</code> / <code>commit</code>)	Continuous (Working copy is always a commit)
The "Index"	Staging area is central to the workflow	No staging area (everything is tracked)
Conflicts	Blocking: Must be resolved to continue	Data: Stored in the commit; fix them later
Branches	Named branches are required to track work	Anonymous (use "bookmarks" only for remotes)
Safety Net	<code>git reflog</code> (hard to parse)	<code>jj undd</code> (first-class, instant rollback)
Rebasing	Destructive & iterative	Transformative & automatic
Remote Sync	Pushes/Pulls branches	Syncs bookmarks to Git remotes
Querying	CLI flags (<code>--grep</code> , <code>--author</code>)	Functional language (Revsets)

...who knows, maybe I can present jujutsu in another talk!!!

🔗 Links to other third generation VCS tools

- Jujutsu VCS: <https://www.jj-vcs.dev/latest>
- Darcs: <https://darcs.net/>
- Pijul: <https://pijul.org/>
- Sapling: <https://sapling-scm.com/>

Q&A

