

• MASTERING • GITHUB

GitHub is more than just a File Store

GitHub is a tool for ...

- Sharing
- Discussing
- Reviewing your code

In this course we'll be focusing on
best practices for using GitHub

Prerequisites

You should be familiar with ..

- Adding and committing files
- Branching
- Merging
- Rebasing
- Handling conflicts

If you're not ...

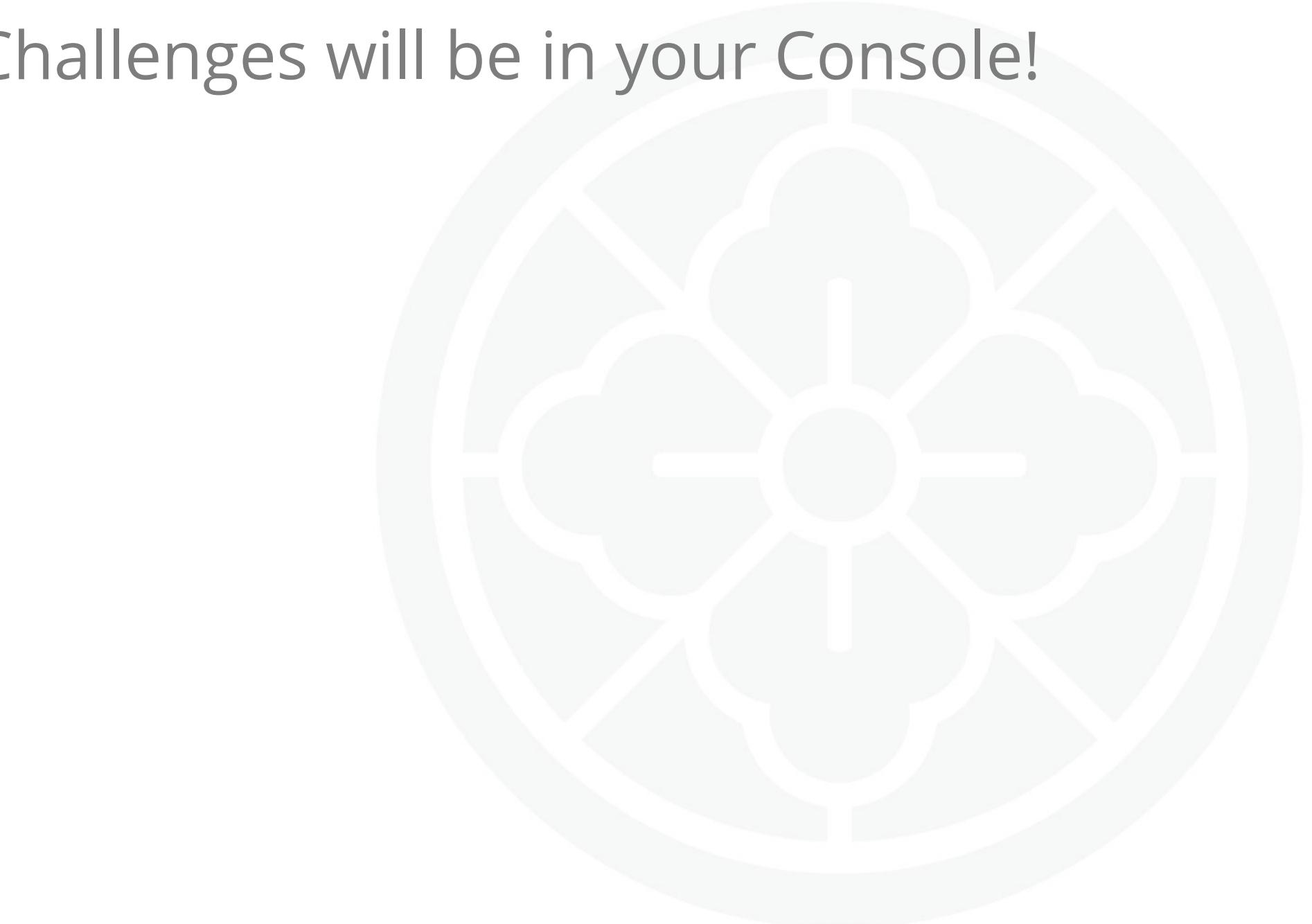


Git Configuration

Level 1: Section 1

Please Follow Along

The Challenges will be in your Console!



Different Levels of Configuration

--local

to set configuration for a single repo

--global

to set configuration for your user

--system

to set configuration for all users

Examples:

set local email if using a different (personal) GitHub account

```
$ git config --local user.email bride@revenge.org
```

colorizing display for **all** users of the system

```
$ git config --system color.ui true
```

Checking Your Global Configuration

To check the global username

```
$ git config --global user.name
```

Bill

To check the global email

```
$ git config --global user.email
```

bill@deadly-vipers.com

Setting Your Global Configuration

To set the global username

```
$ git config --global user.name "Beatrix Kiddo"
```

To set the global email

```
$ git config --global user.email beatrix@deadly-vipers.com
```

Viewing Global Configurations

```
$ git config --global --list
```

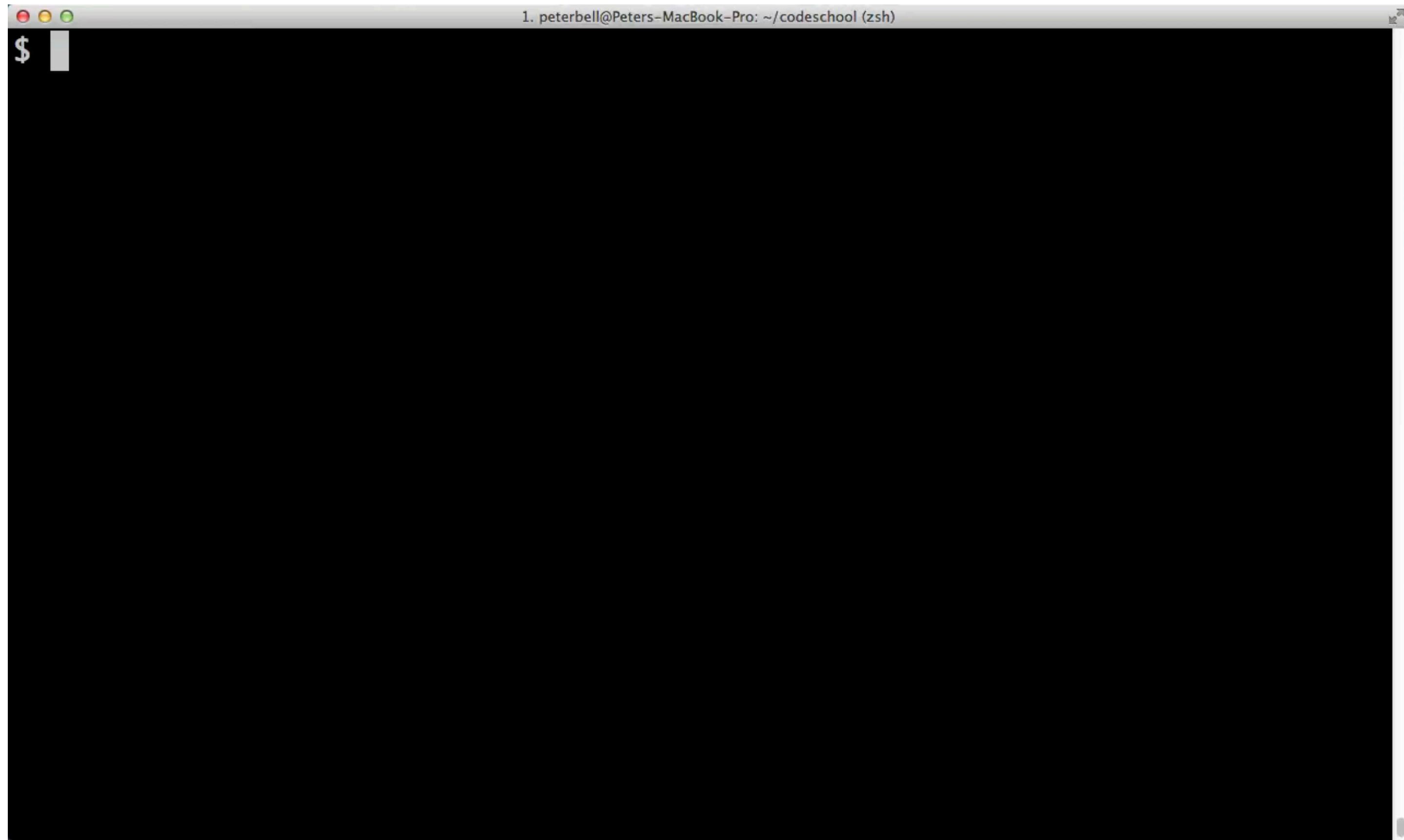
```
user.name=Beatrix Kiddo
user.email= beatrix@deadly-vipers.com
core.autocrlf=input
color.ui=true
```

```
$ cat ~/.gitconfig
```

```
[user]
  name = Beatrix Kiddo
  email = beatrix@deadly-vipers.com
[core]
  autocrlf = input
[color]
  ui = true
```

This file can be edited using a text editor.

Viewing Global Configurations



Viewing Local Configurations

This only works within a git repository.

```
$ git config --local --list
```

```
user.name=Beatrix Kiddo
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
core.ignorecase=true
core.precomposeunicode=false
```

```
$ cat .git/config
```

```
[user]
    name = Beatrix Kiddo
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
    ignorecase = true
    precomposeunicode = false
```

This file can be edited just like the global `.gitconfig`.

Configuring Line Endings

Linux/Mac:

```
$ git config --global core.autocrlf input
```

auto carriage return/line feed handling

Windows:

```
$ git config --global core.autocrlf true
```

Both of these strip carriage returns, on saving files to the repository.

“true” adds the CRs back in when you check files out to the working directory, making them easier to edit in some Windows apps that require CRs.

Configuring Push Default

Matching - Pushes all matching branches up to GitHub

Simple - Just pushes current branch up to GitHub

```
$ git config --global push.default simple
```



the default in Git 2.0 and later

Configuring Pull Default

git pull

- git fetch
- git merge

useless merge

```
* | 45db1fb (HEAD, master) Merge branch 'master' of  
github.com:peterbell/test  
|\  
| * 159649f (origin/master, origin/HEAD) Added home  
page  
* | d0b8d0b Added about us page  
|/  
* 2798de1 Initial commit
```



git pull --rebase

- git fetch
- git rebase

```
* 3bb211f (HEAD, master) Added about us page  
* 159649f (origin/master, origin/HEAD) Added home page  
* 2798de1 Initial commit
```



We taught rebasing in Git Real 2.

Configuring Pull Default

```
* 3bb211f (HEAD, master) Added about us page  
* 159649f (origin/master, origin/HEAD) Added home page  
* 2798de1 Initial commit
```

Defaults all new branches to fetch then rebase

```
$ git config --global pull.rebase true
```

Configuring Reuse Recorded Resolution (ReReRe)

Records all fixes to merge conflicts

Reuses them automatically if the same conflict recurs

```
$ git config --global rerere.enabled true
```

Particularly useful when cherry picking to multiple branches or constantly rebasing.

Configuring Output Colors

Colorizes output to improve readability

```
$ git config --global color.ui true
```

Default from Git 1.8.4 onwards, so only needed for older versions.

Configuring Aliases

Configuring Aliases - "git s"

```
$ git status
```

```
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    modified:   this.txt

Untracked files:
  (use "git add <file>..." to include in what
will be committed)

    that.txt
```



```
$ git status -s
```

```
M this.txt
?? that.txt
```



Overly verbose!

```
$ git config --global alias.s "status -s"
```

Now we can just type "git s" to get the (silent) status output.

Configuring Aliases - "git lg"

```
$ git log
```

```
commit 14a1cb7b6880a95af739b992c5c7248d858941f2
```

```
Author: Yves Senn <yves.senn@gmail.com>
```

```
Date: Mon Jun 2 16:40:33 2014 +0200
```

```
docs, fix typo [ci skip]
```

```
commit d7ba8b70183a22cc505fe1735c2c2ca3e1c0f174
```

```
Author: Yves Senn <yves.senn@gmail.com>
```

```
Date: Mon Jun 2 13:21:12 2014 +0200
```

```
test pg, add basic test cases for point datatype.
```



```
$ git lg
```

```
* 14a1cb7 (HEAD, origin/master, origin/HEAD, master)
docs, fix typo [ci skip]
* d7ba8b7 test pg, add basic test cases for point datatype.
* f17fa8e pg, inline casting methods into `OID::Type` objects.
* 4c66ab2 Feature requests should be made on the mailing list, not submitted to the issue tracker. See also #15455 [ci skip]
* fab8d07 Merge pull request #15444 from sgrif/sgyaml-new-record
\ 
| * e08494a New records should remain new after yaml serialization
|/
```



Overly verbose!

```
$ git config --global alias.lg "log --oneline --decorate --all --graph"
```

"git lg" will now give us a concise, useful log output.

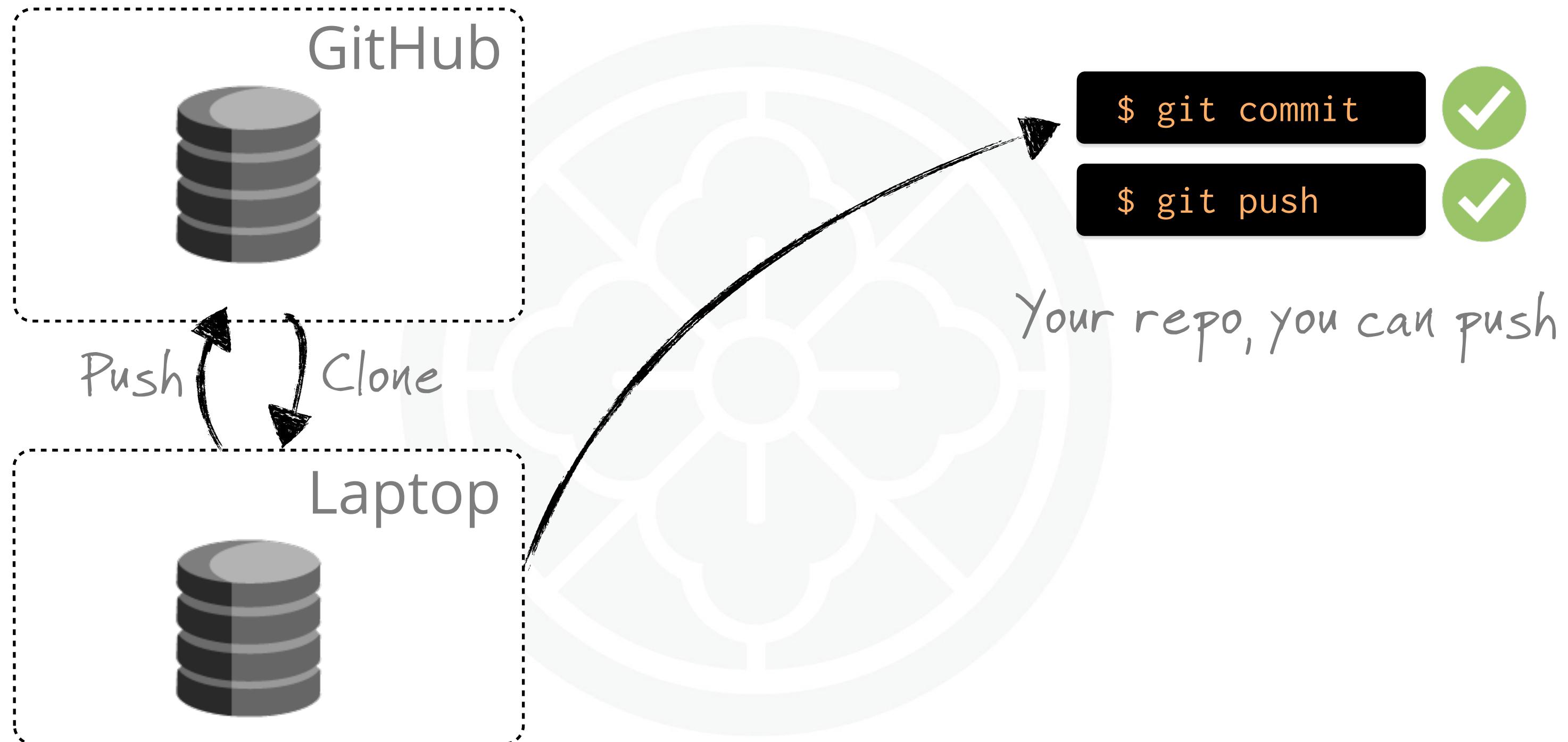
Sharing Configurations

- Wiki page
- Put dot files into a repo

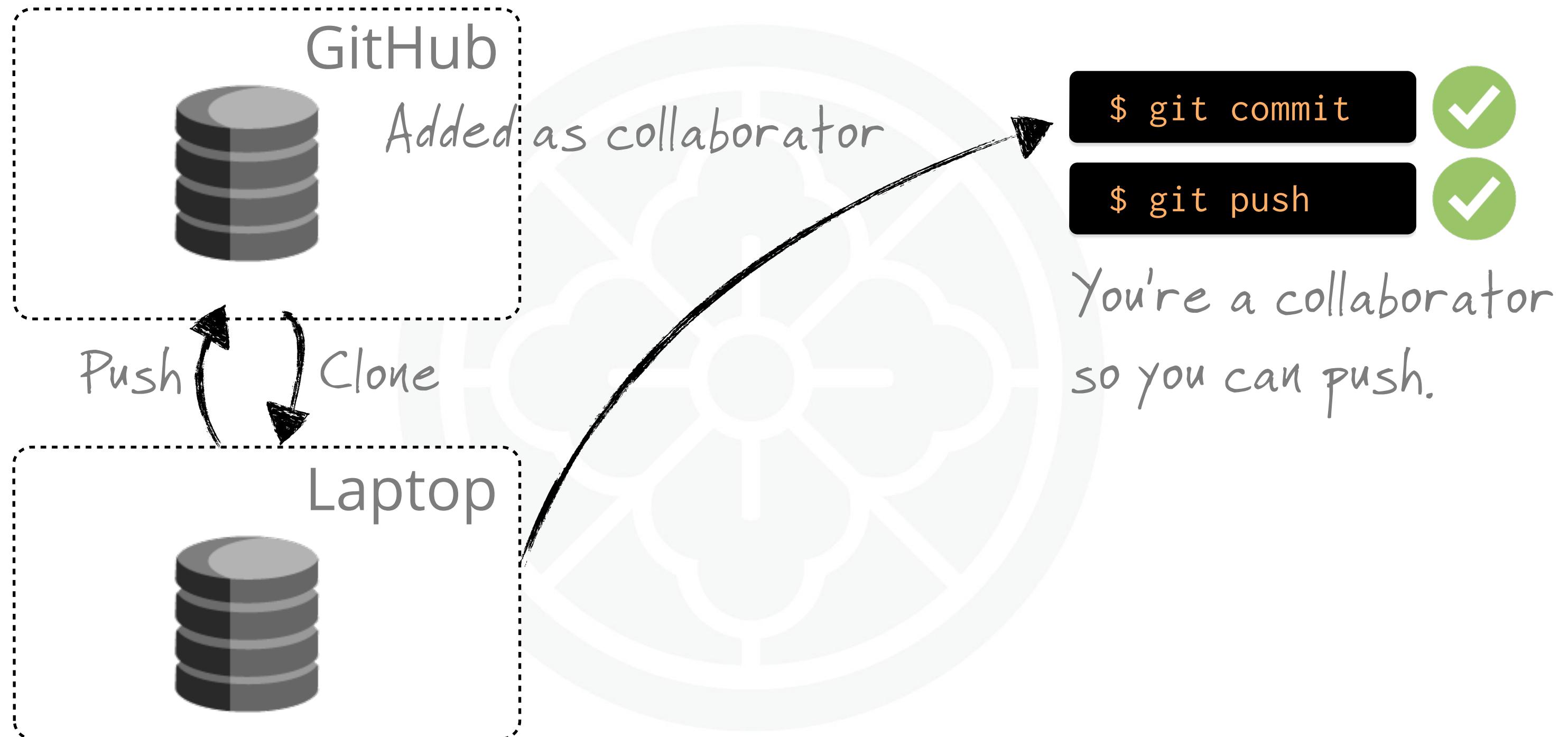
Forking and Cloning

Level 1: Section 2

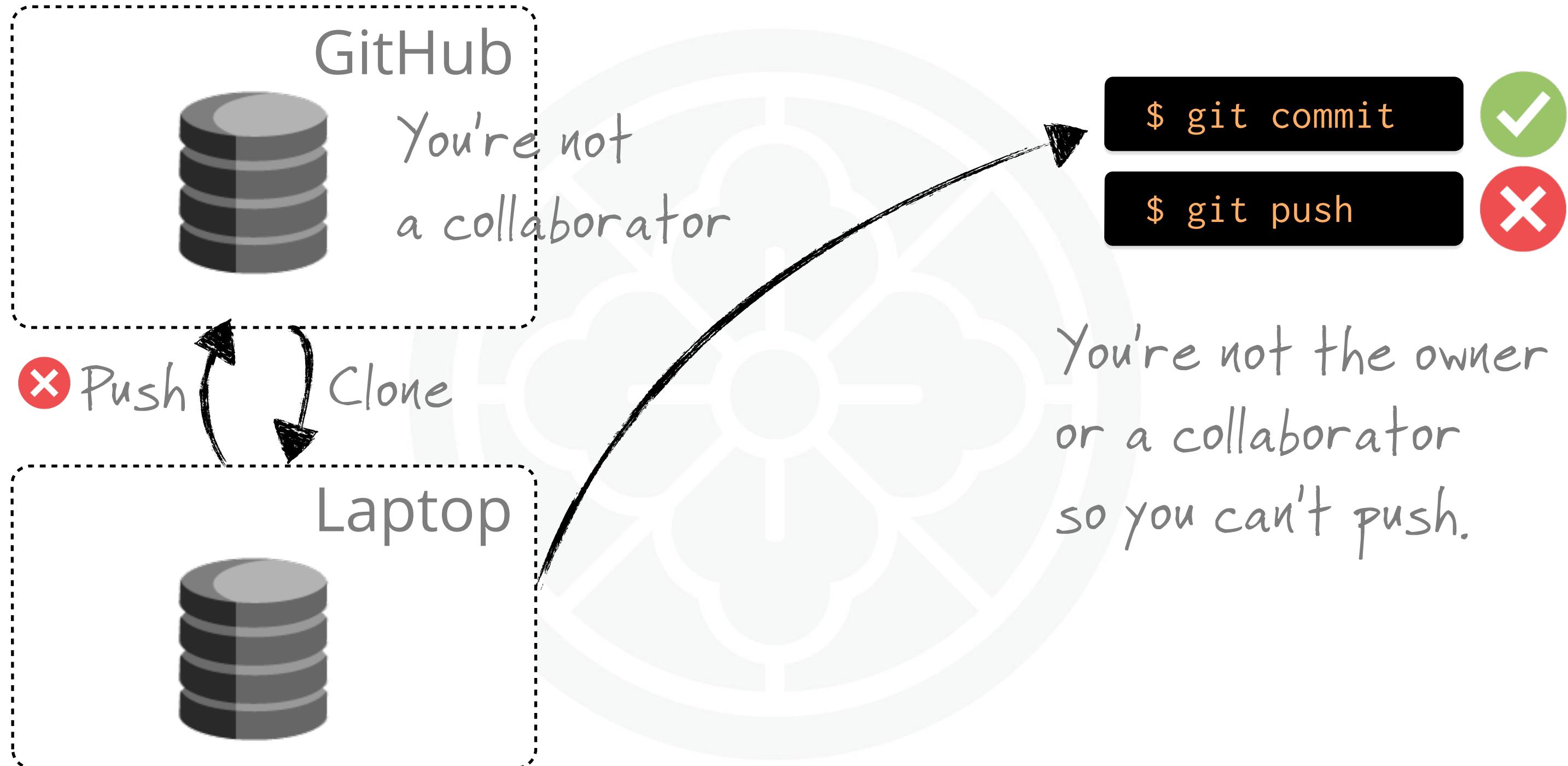
Creating a Repo



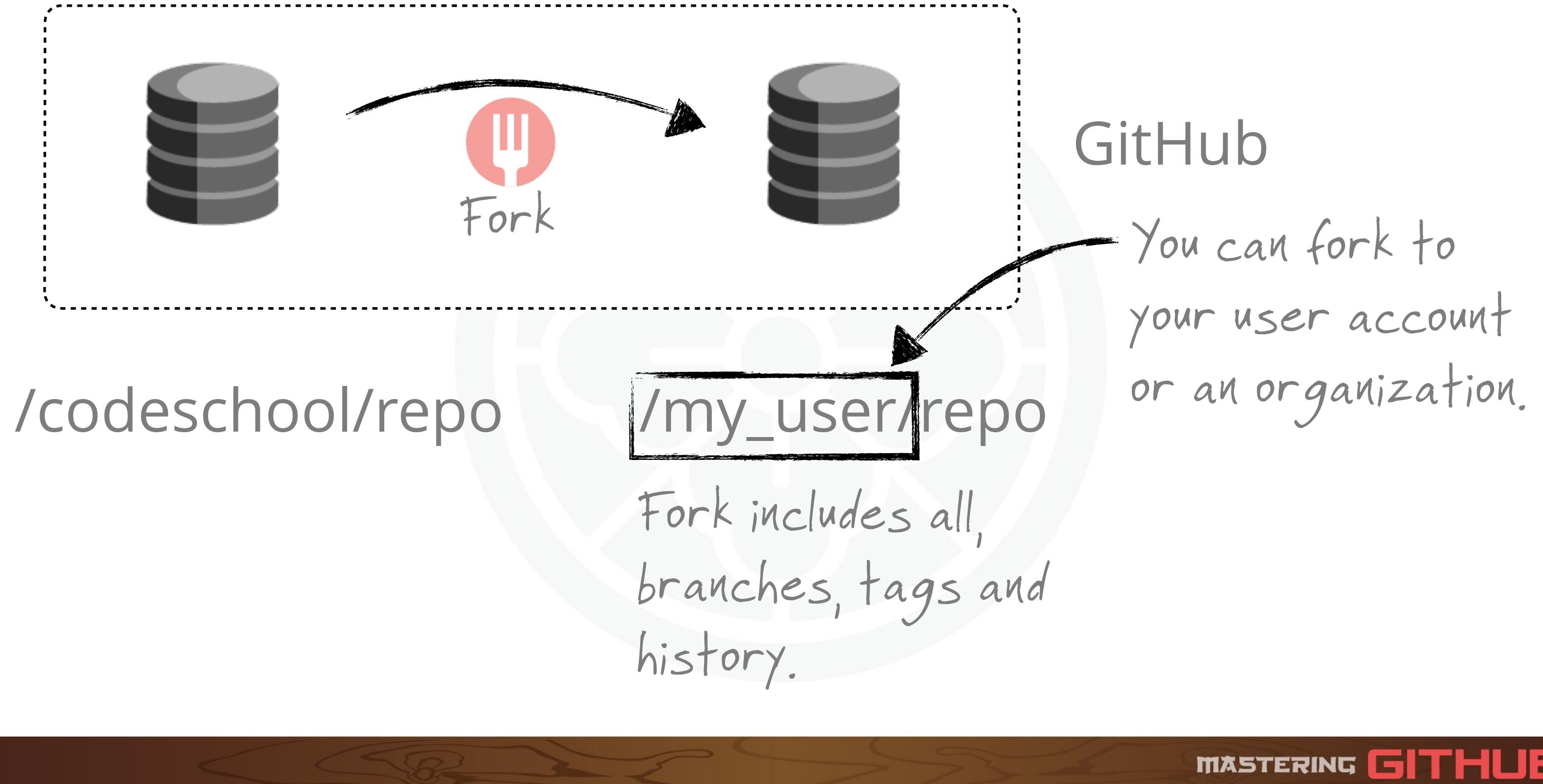
Cloning a Repo as a Collaborator



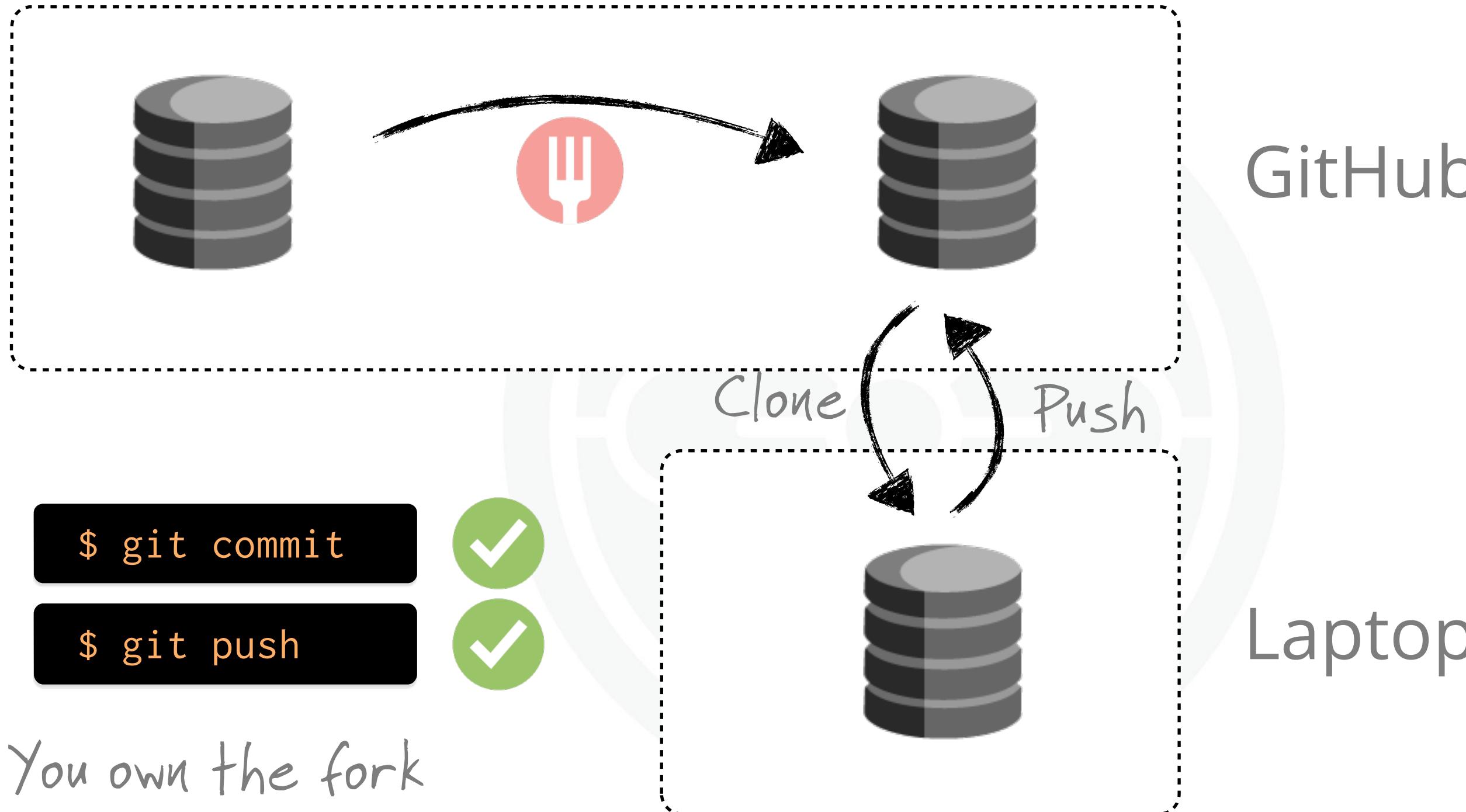
Cloning a Repo if You're Not a Collaborator



Contributing to Projects via Forking



Forking and Cloning



deadlyvipers/dojo_rules

GitHub, Inc. [US] https://github.com/deadlyvipers/dojo_rules

This repository Search or type a command Explore Gist Blog Help PeterBell + × ↗

deadlyvipers / dojo_rules

Unwatch 1 Star 0 Fork 0

Description Website

Short description of this repository Website for this repository (optional) Save or cancel

2 commits 1 branch 0 releases 0 contributors

branch: master +

dojo_rules / +

Added README to the project

Beatrix Kiddo authored Tuesday at 8:05pm latest commit 648a2dfab4

README.md Added README to the project 2 days ago

dojo_rules.md Initial dojo rules 2 days ago

README.md

Dojo Rules

Issues 0

Pull Requests 0

Wiki

Pulse

Graphs

Network

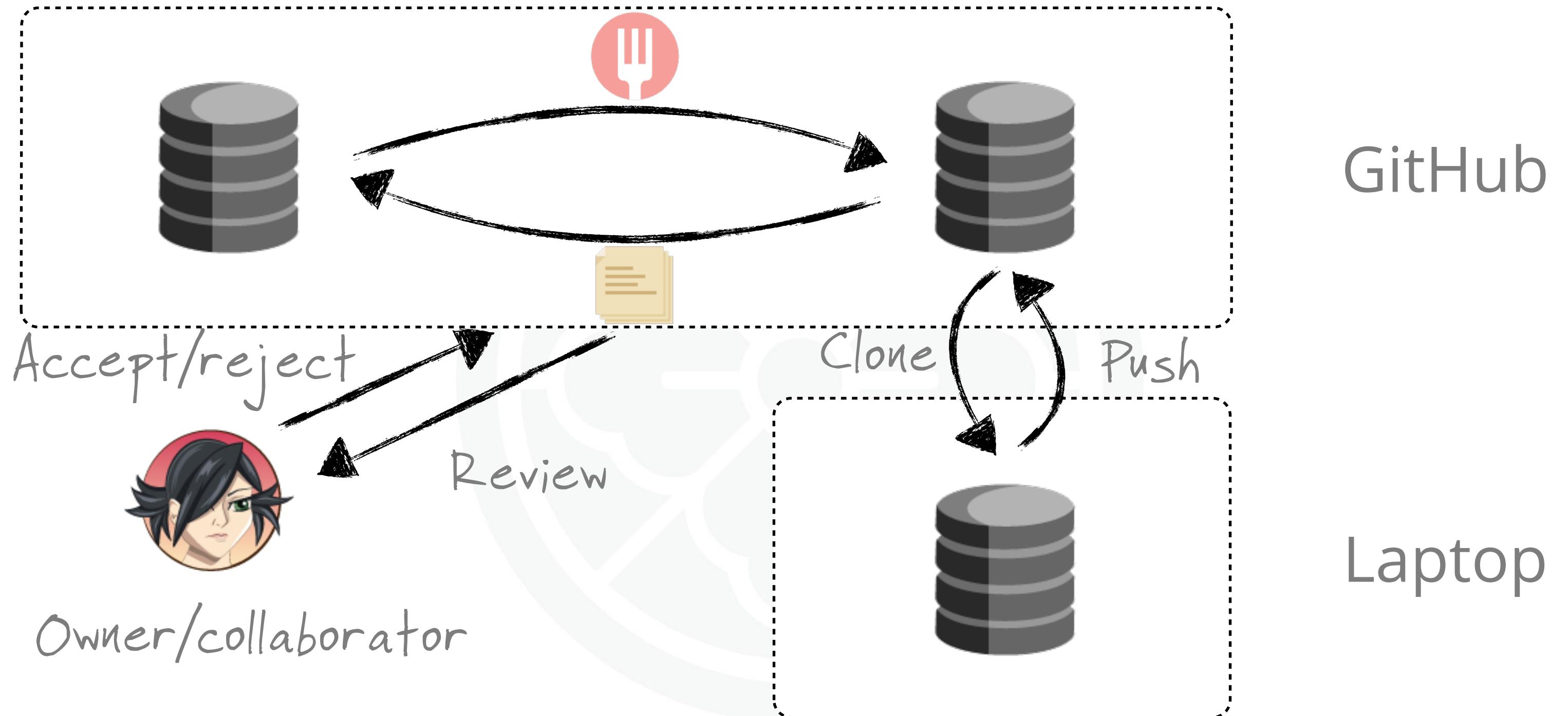
Settings

HTTPS clone URL

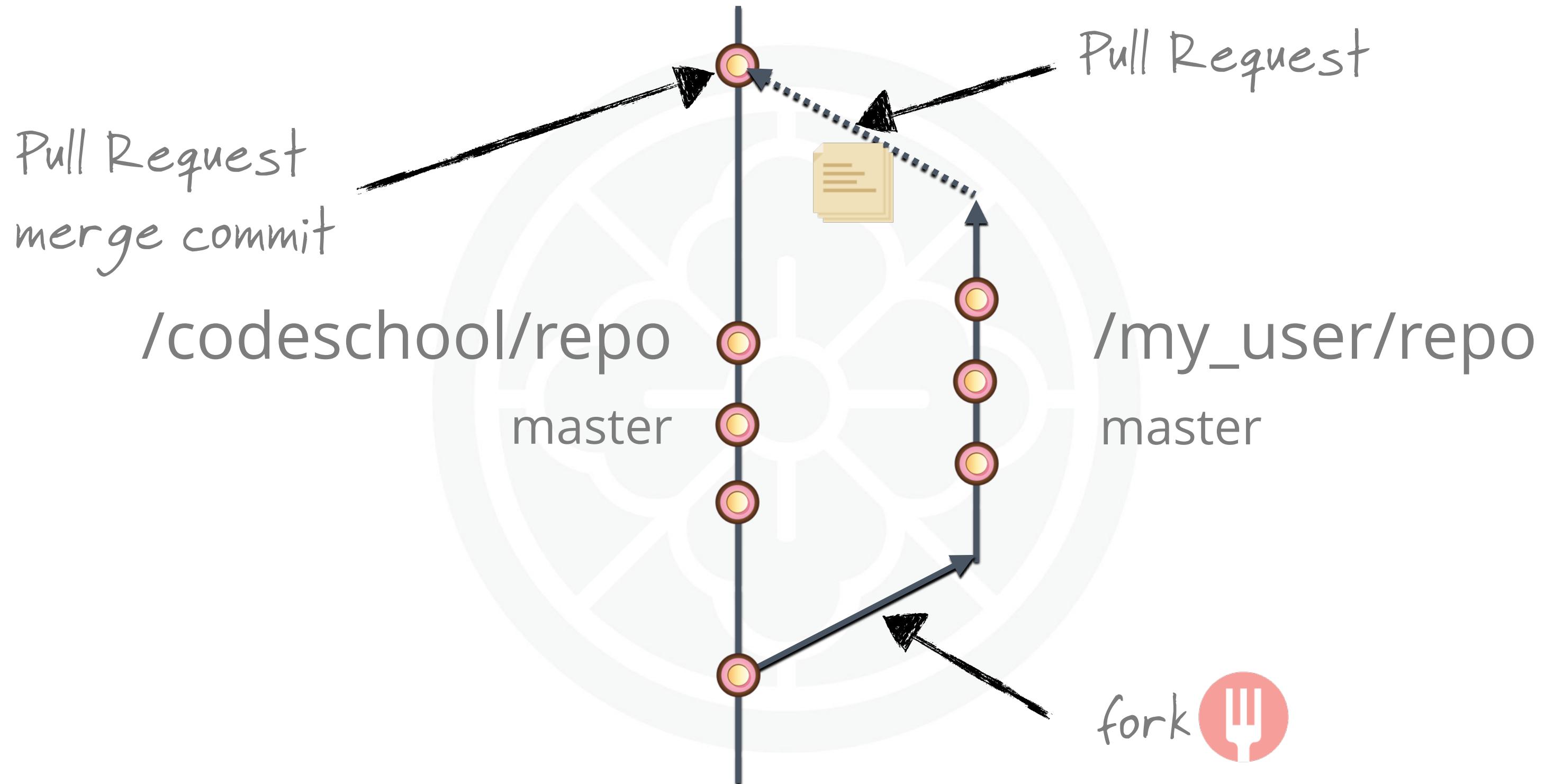
Submitting a Pull Request

Level 1: Section 3

Submitting a Pull Request



Reviewing a Pull Request



PeterBell/dojo_rules

GitHub, Inc. [US] https://github.com/PeterBell/dojo_rules

This repository Search or type a command Explore Gist Blog Help PeterBell + ×

PUBLIC PeterBell / dojo_rules forked from deadlyvipers/dojo_rules

Unwatch 1 Star 0 Fork 1

Description Website

Short description of this repository Website for this repository (optional) Save or cancel

3 commits 1 branch 0 releases 0 contributors

branch: master + dojo_rules / +

This branch is 0 commits ahead and 0 commits behind master Pull Request Compare

Not every Sensei deserves to be respected

PeterBell authored a minute ago latest commit a4b95d10b0

README.md Added README to the project 6 days ago

dojo_rules.md Not every Sensei deserves to be respected a minute ago

README.md

Dojo Rules

This repository contains a list of dojo rules for the Deadly Vipers dojo

Pulse Graphs Network Settings

HTTPS clone URL https://github.com/ You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop Download ZIP

Reviewing a Pull Request

◀ Most Senseis should be respected #2

Merged PeterBell merged 1 commit into `deadlyvipers:master` from `githubteacher:master` about 22 hours ago

Conversation 0

Commits 1

Files changed 1



githubteacher commented 22 hours ago

No description provided.

Most Senseis should be respected

e42abab

PeterBell merged commit `022ac13` into `deadlyvipers:master` from `githubteacher:master`
22 hours ago

PeterBell closed this 22 hours ago

Anyone can comment on a PR

They can also checkout your branch, commit and add their changes to your PR.

The owner or any collaborator can merge in your PR.

You could merge in your own PR without review - just don't do it!

deadlyvipers/dojo_rules

GitHub, Inc. github.com/deadlyvipers/dojo_rules

Reader

http://skinn...worlds.dev/ Apple Yahoo! Google Maps YouTube Wikipedia News Popular

This repository Search or type a command Explore Gist Blog Help

githubteacher + - X E

PUBLIC deadlyvipers / **dojo_rules**

Watch 1 Star 0 Fork 1

2 commits 1 branch 0 releases 0 contributors

branch: master +

Added README to the project

Beatrix Kiddo authored Tuesday at 8:05pm latest commit 648a2dfab4

README.md Added README to the project 6 days ago

dojo_rules.md Initial dojo rules 6 days ago

README.md

Dojo Rules

This repository contains a list of dojo rules for the Deadly Vipers dojo

Code Issues 1 Pull Requests 1 Wiki Pulse Graphs Network

HTTPS clone URL <https://github.com/>

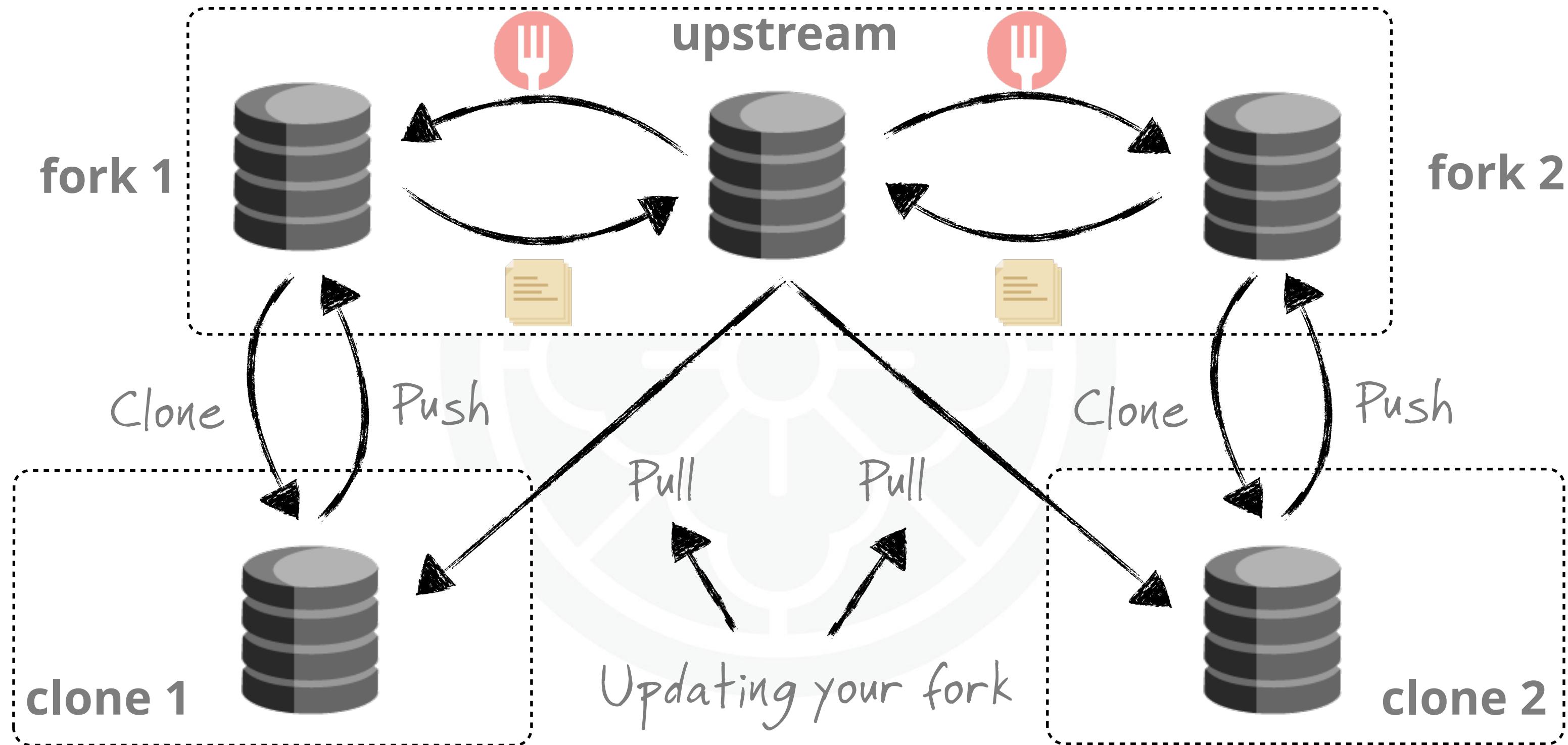
You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop Download ZIP

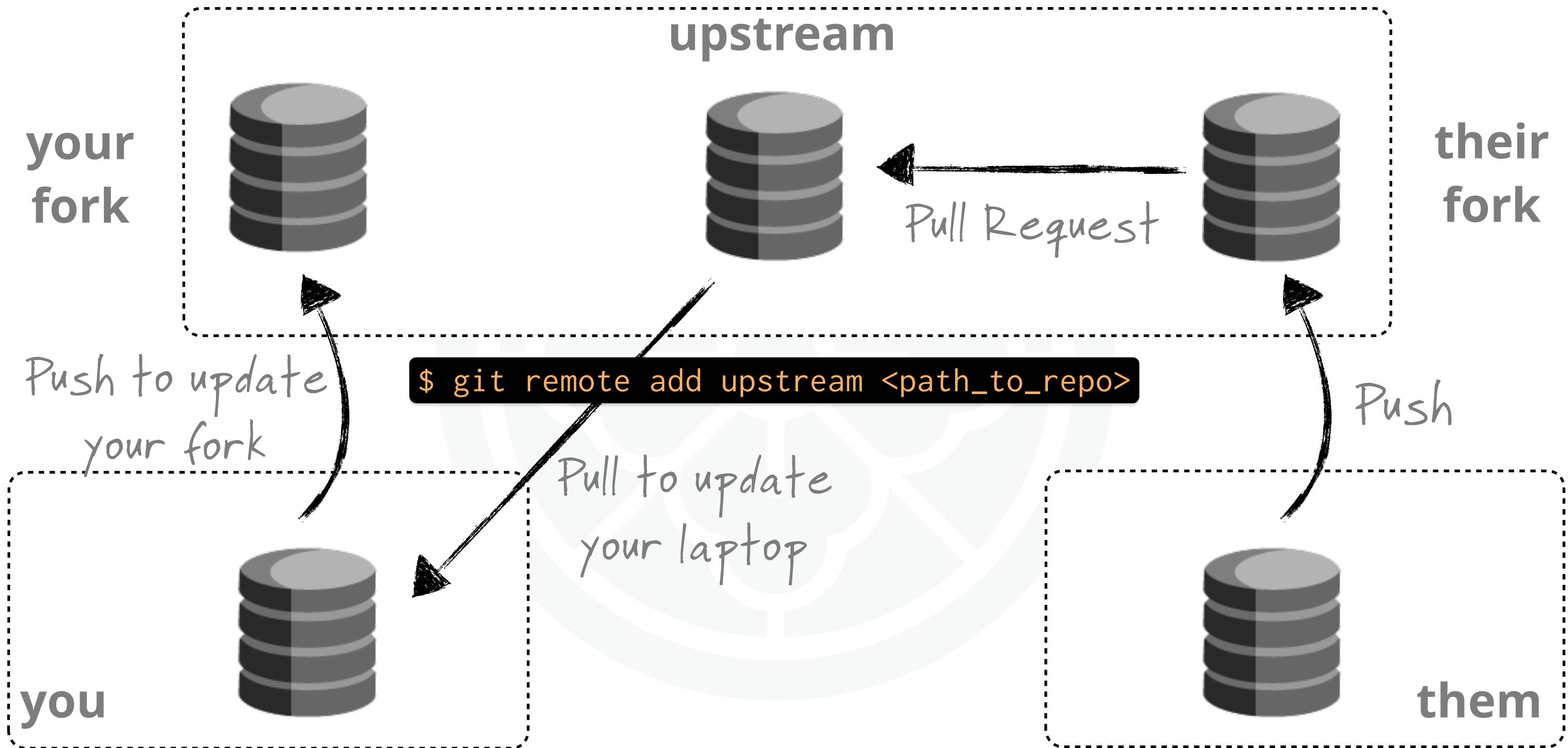
Updating your Fork

Level 1: Section 4

Updating your Fork



Updating your Fork



Updating your Fork

Add remote for upstream

```
$ git remote add upstream <path_to_repo>
```

Fetch changes

```
$ git fetch upstream
```

Merge them into master

```
$ git merge upstream/master master
```

Push them to your remote

```
$ git push origin master
```

Updating your Fork

The screenshot shows a GitHub fork repository page for the repository `deadlyvipers/dojo_rules`. The browser address bar shows the URL `github.com/deadlyvipers/dojo_rules`. The GitHub header includes the user icon for `githubteacher`, navigation links for `Explore`, `Gist`, `Blog`, and `Help`, and a search bar. The repository card displays basic statistics: 5 commits, 1 branch, 0 releases, and 0 contributors. A dropdown menu shows the current branch is `branch: master`. Below the stats, a list of commits is shown:

- Merge pull request #1 from PeterBell/master ... by **PeterBell** authored 39 minutes ago latest commit `df1d76604F`
- Added README to the project by **PeterBell** 6 days ago
- You should respect some Senseis by **PeterBell** 39 minutes ago

The `README.md` file content is displayed below:

Dojo Rules

This repository contains a list of dojo rules for the Deadly Vipers dojo

On the right side of the page, there are sidebar links for `Code`, `Issues` (1), `Pull Requests` (1), `Wiki`, `Pulse`, `Graphs`, and `Network`. There is also a section for cloning the repository via `HTTPS` with a URL input field containing `https://github.com/` and a `Clone in Desktop` button, along with a `Download ZIP` button.

• MASTERING • GITHUB

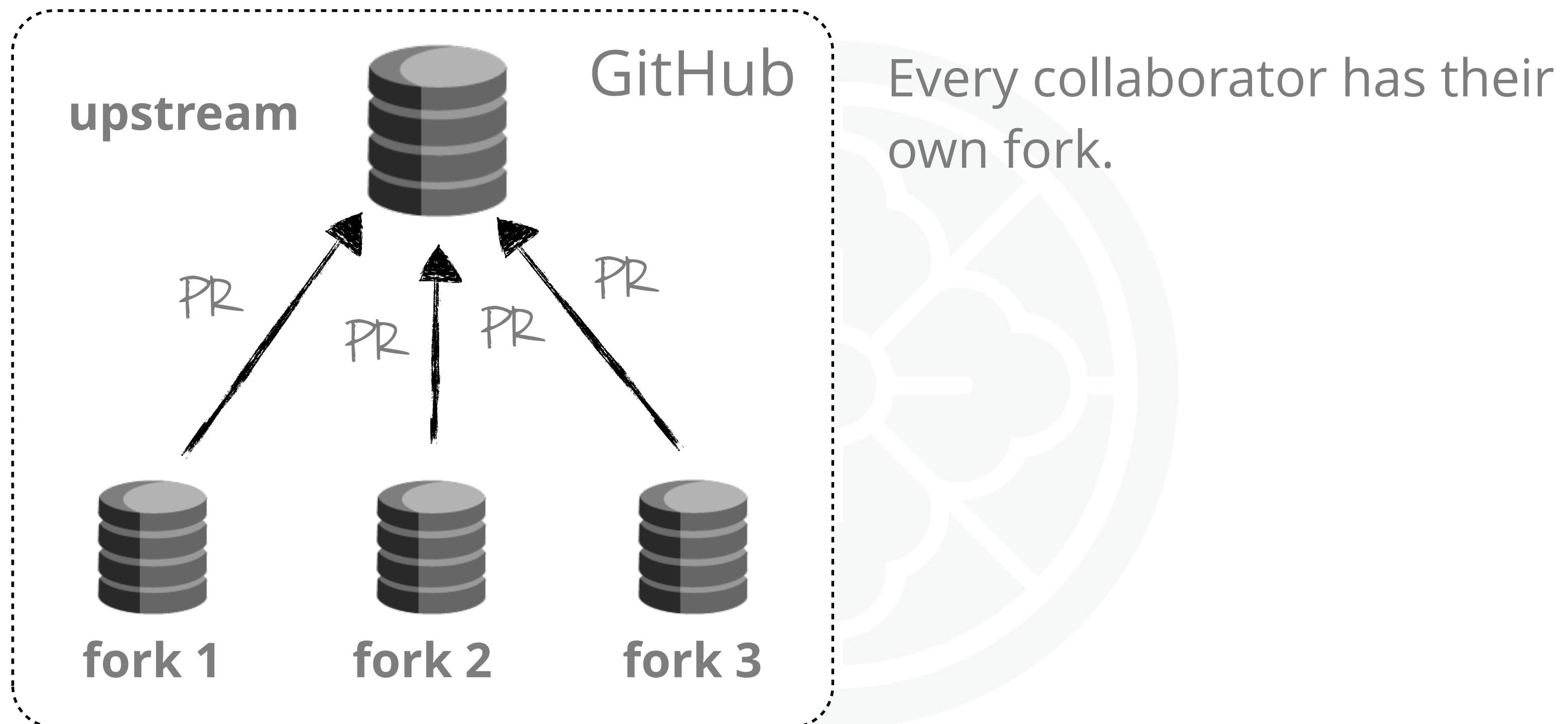


• MASTERING • GITHUB

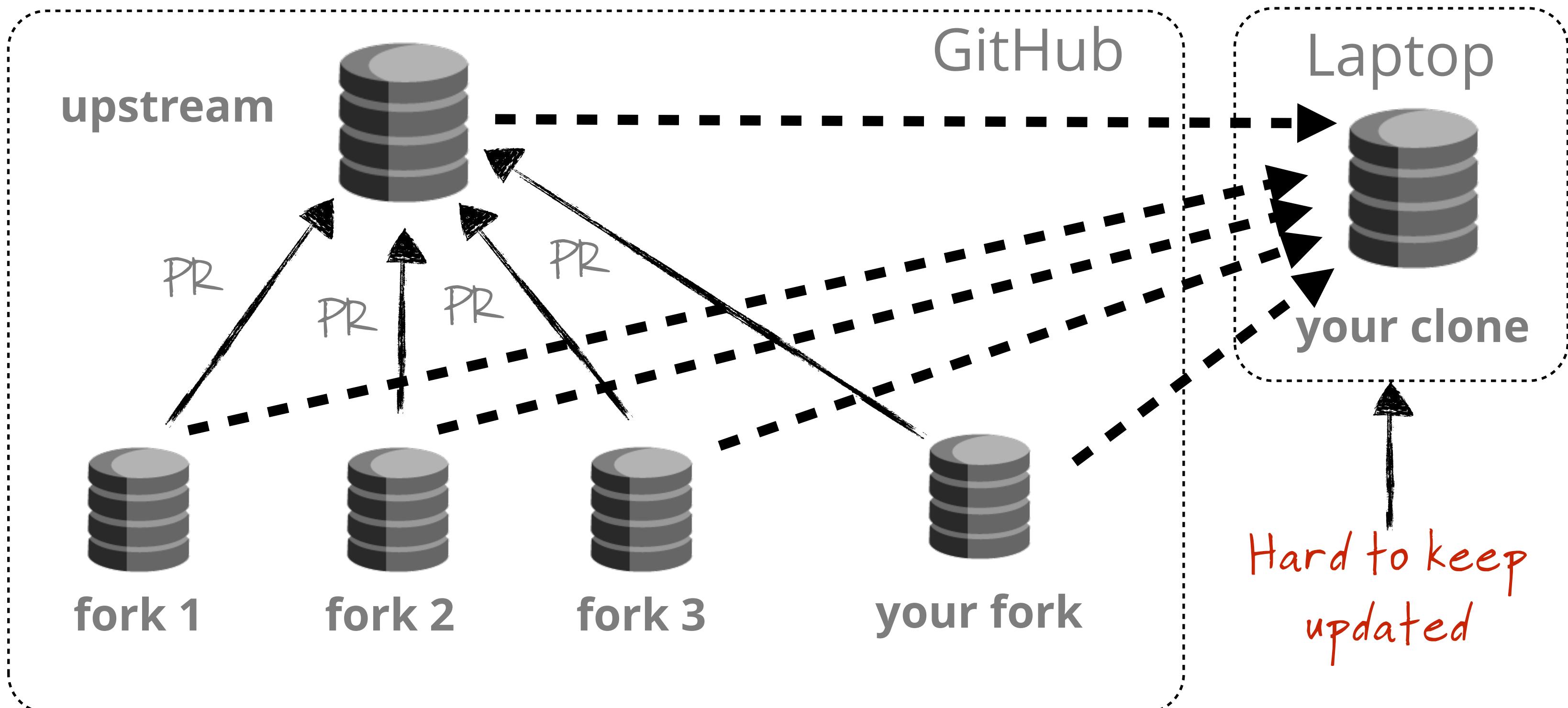
Single Repository Workflows

Level 2: Section 1

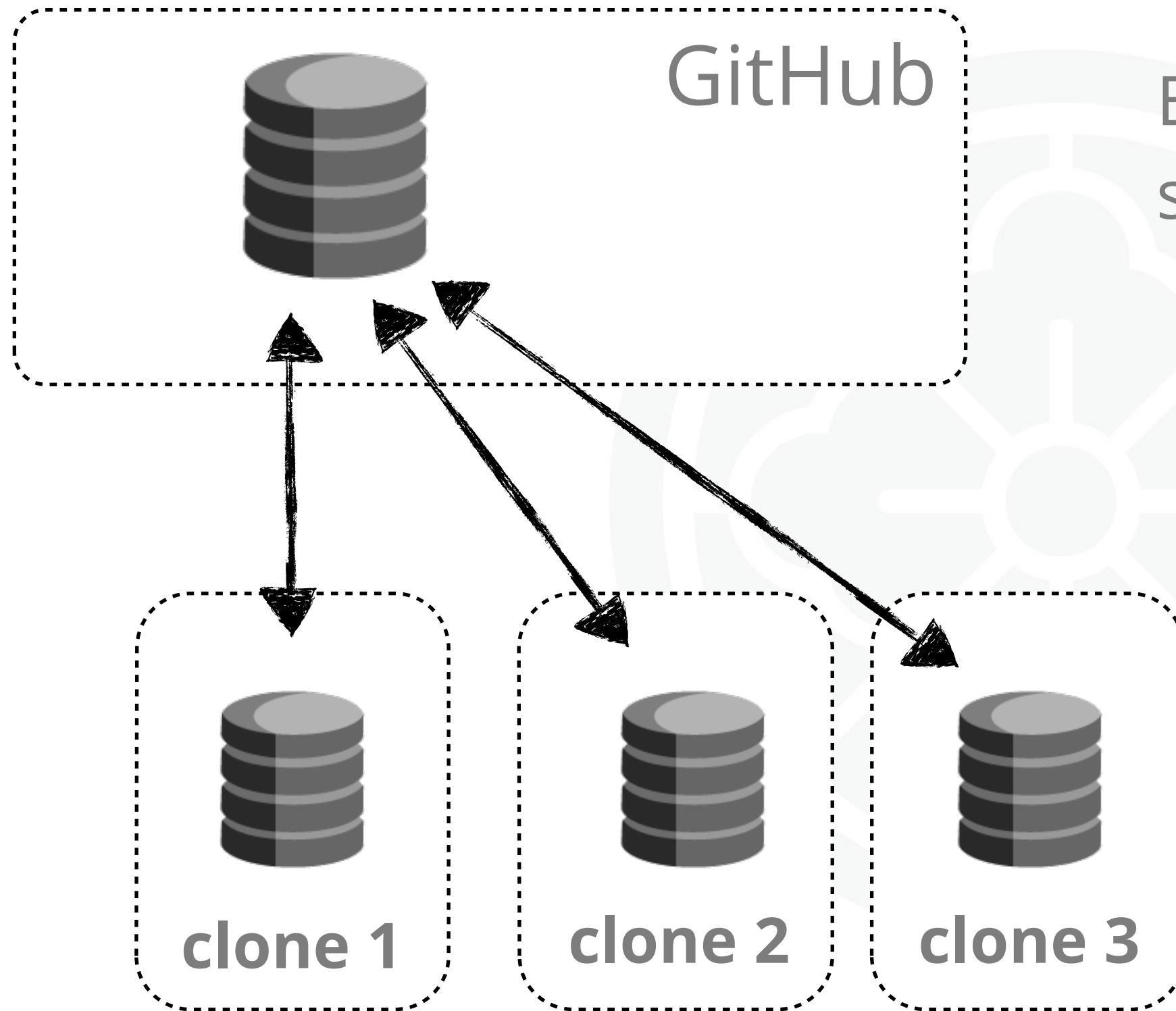
Fork Based Workflow



Fork Based Workflow

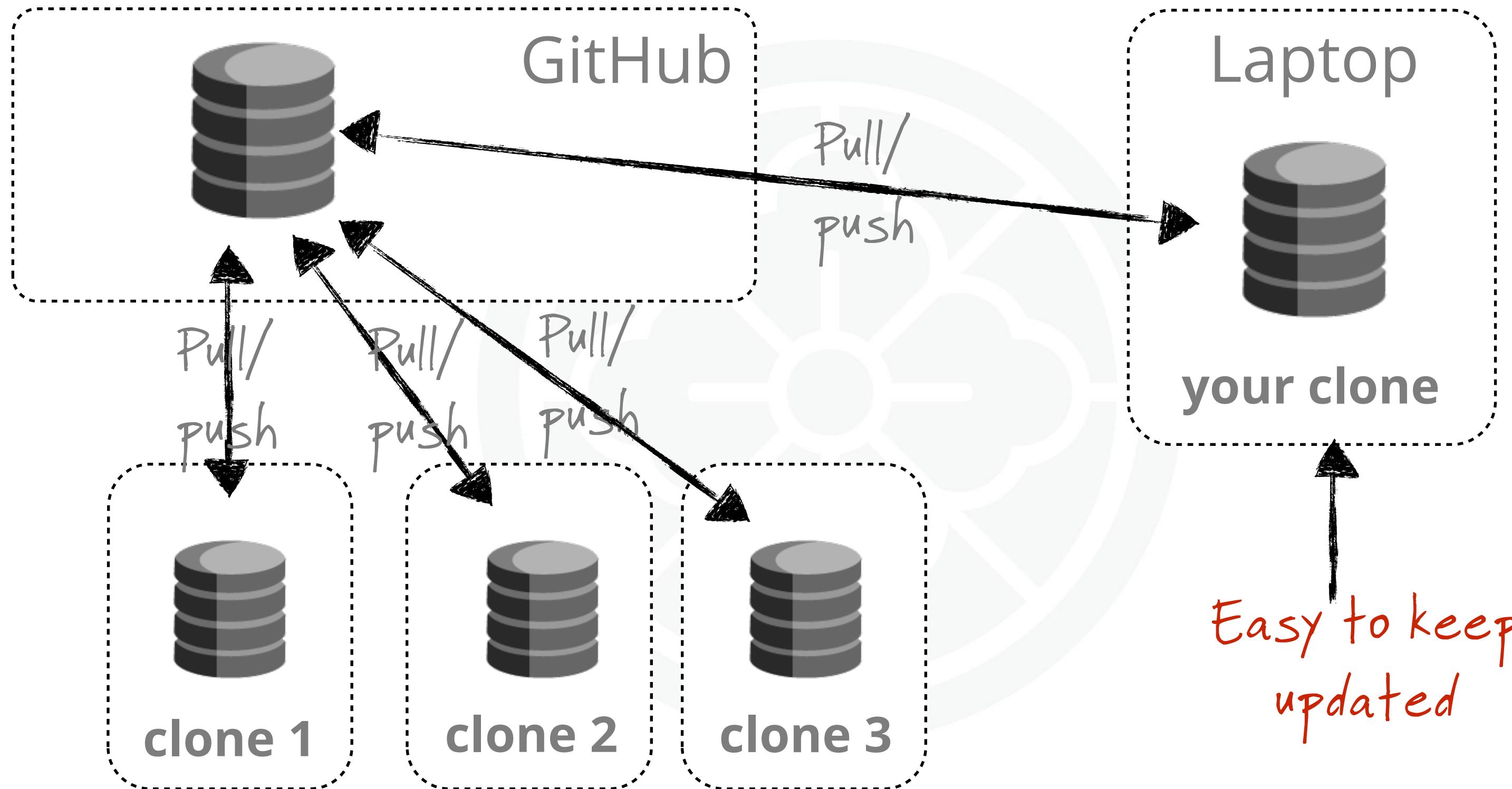


Introducing Single Repository Workflow



Everyone collaborates on a single repo on GitHub.

Introducing Single Repository Workflow

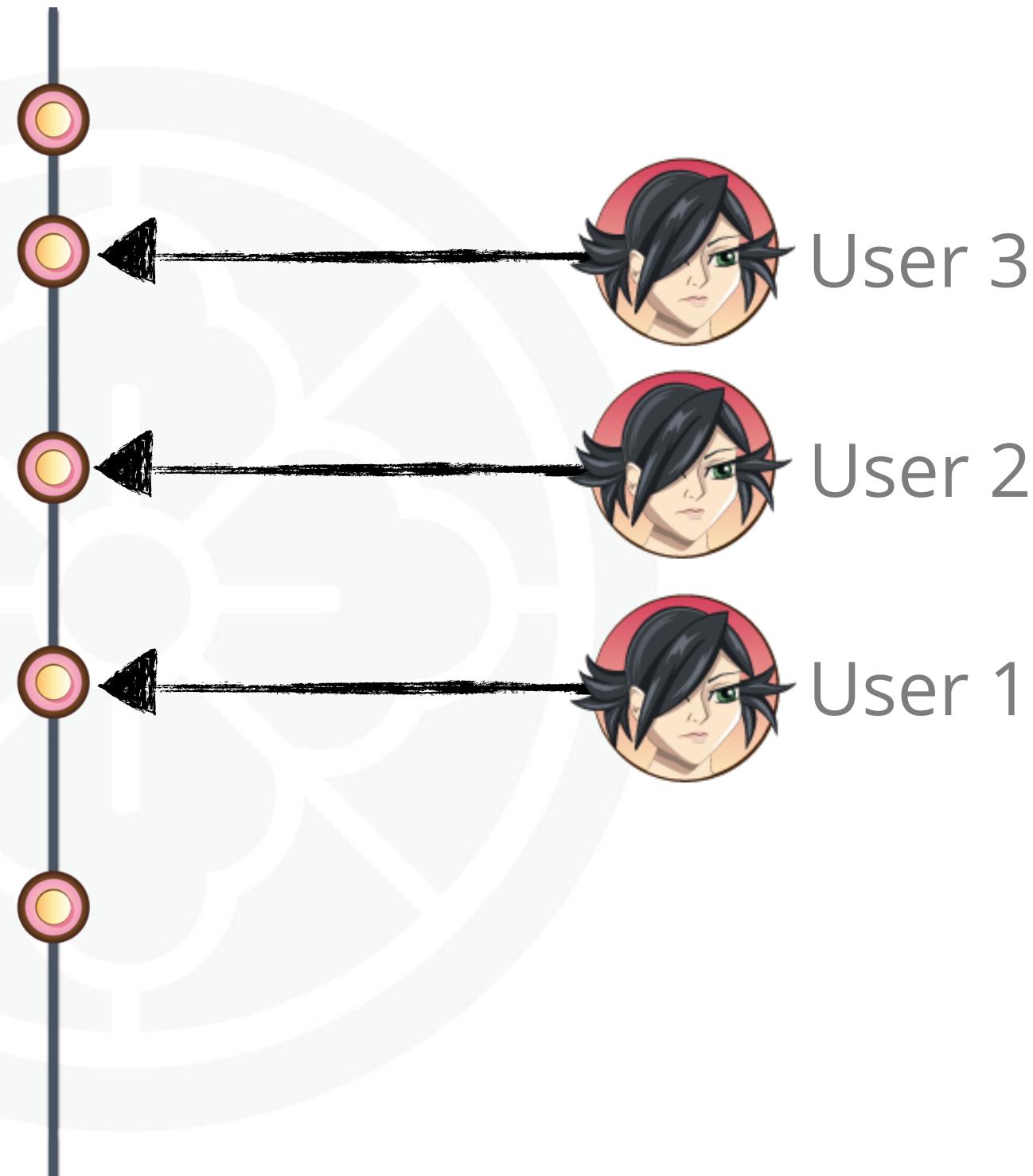


Conflicts if Everyone Commits Directly to Master

/codeschool/repo

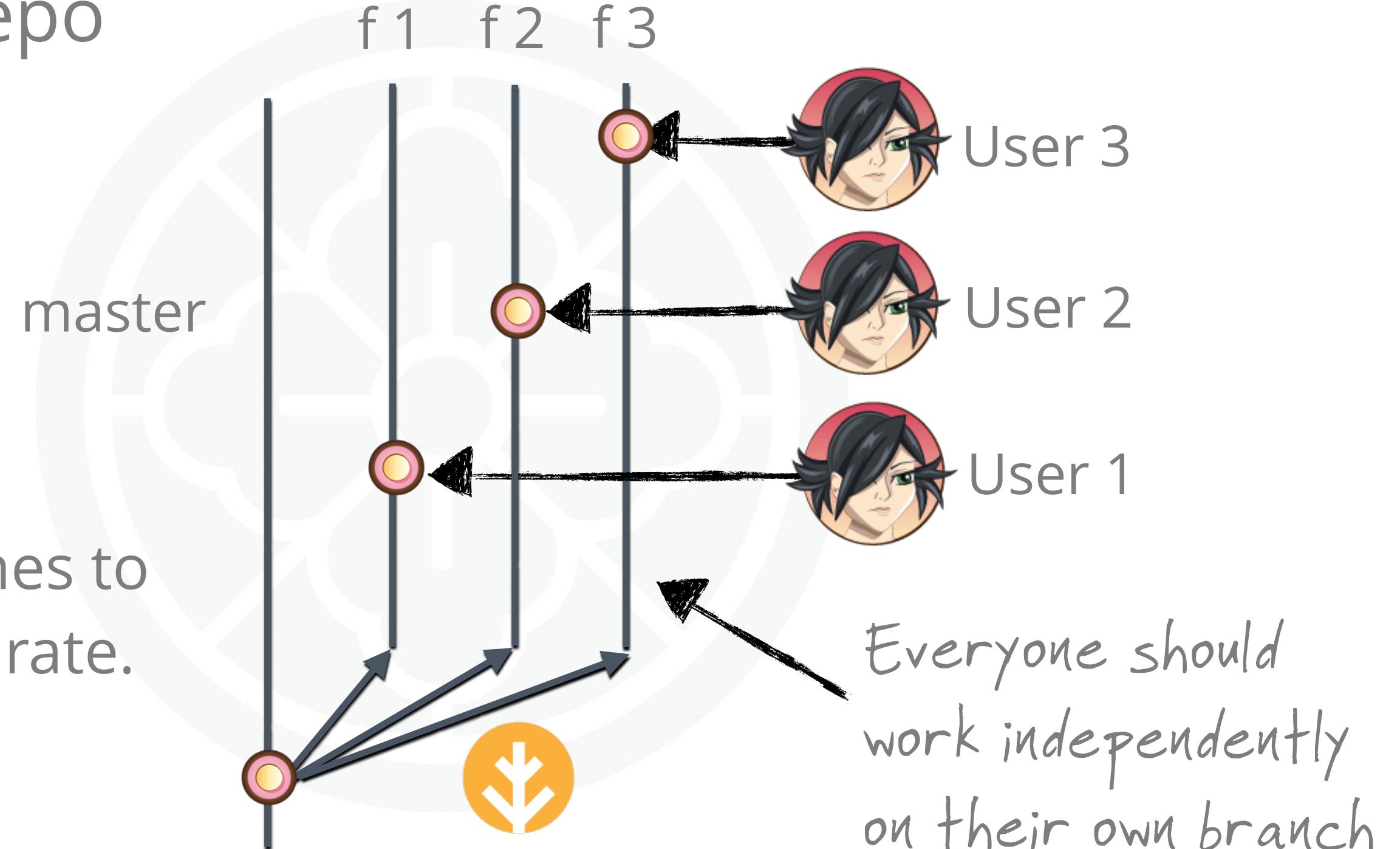
master

If everyone is pushing to
the master branch, you'll
end up stepping on each
other.



Feature branches for Independent Development

/codeschool/repo

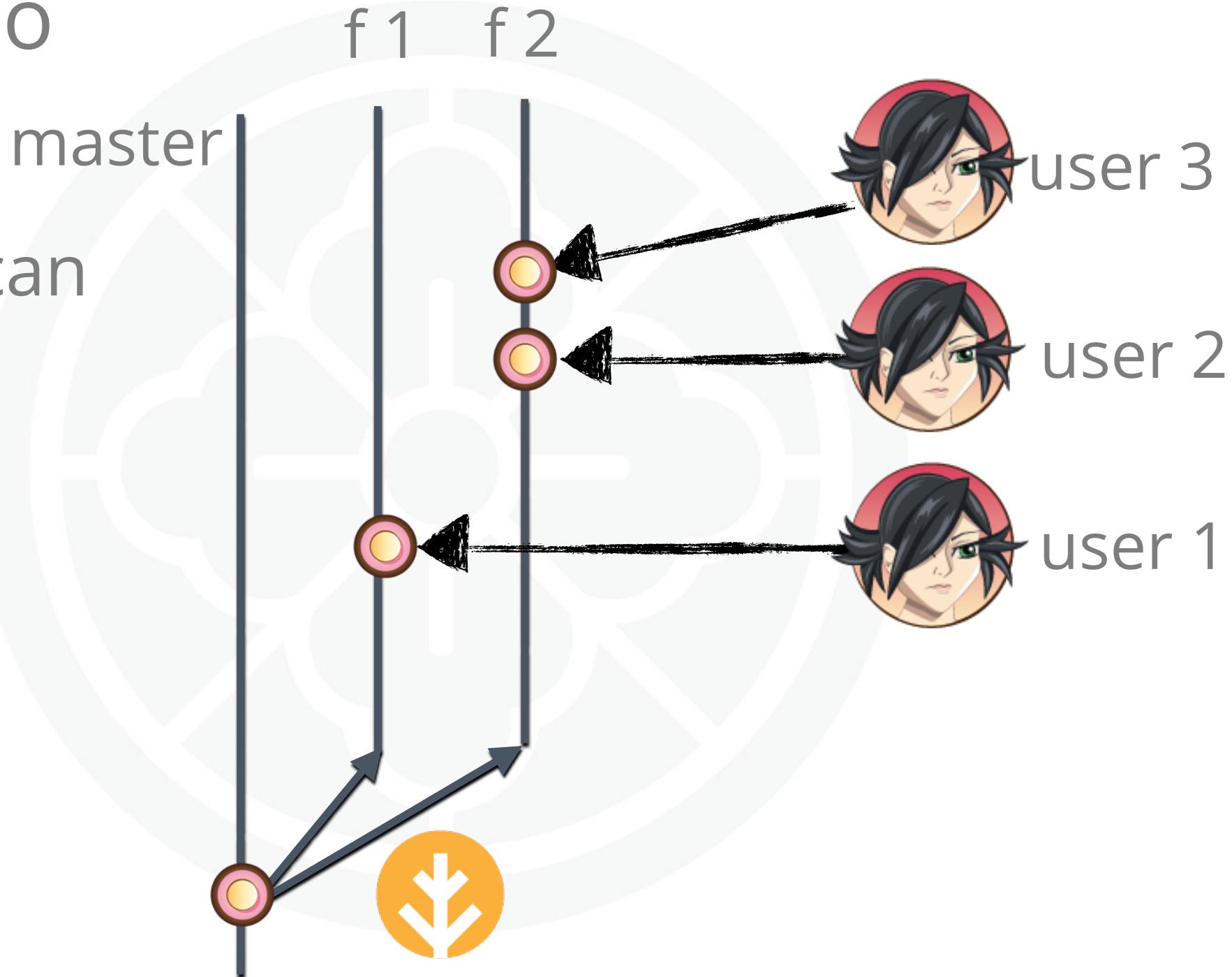


Collaborating on Feature Branches

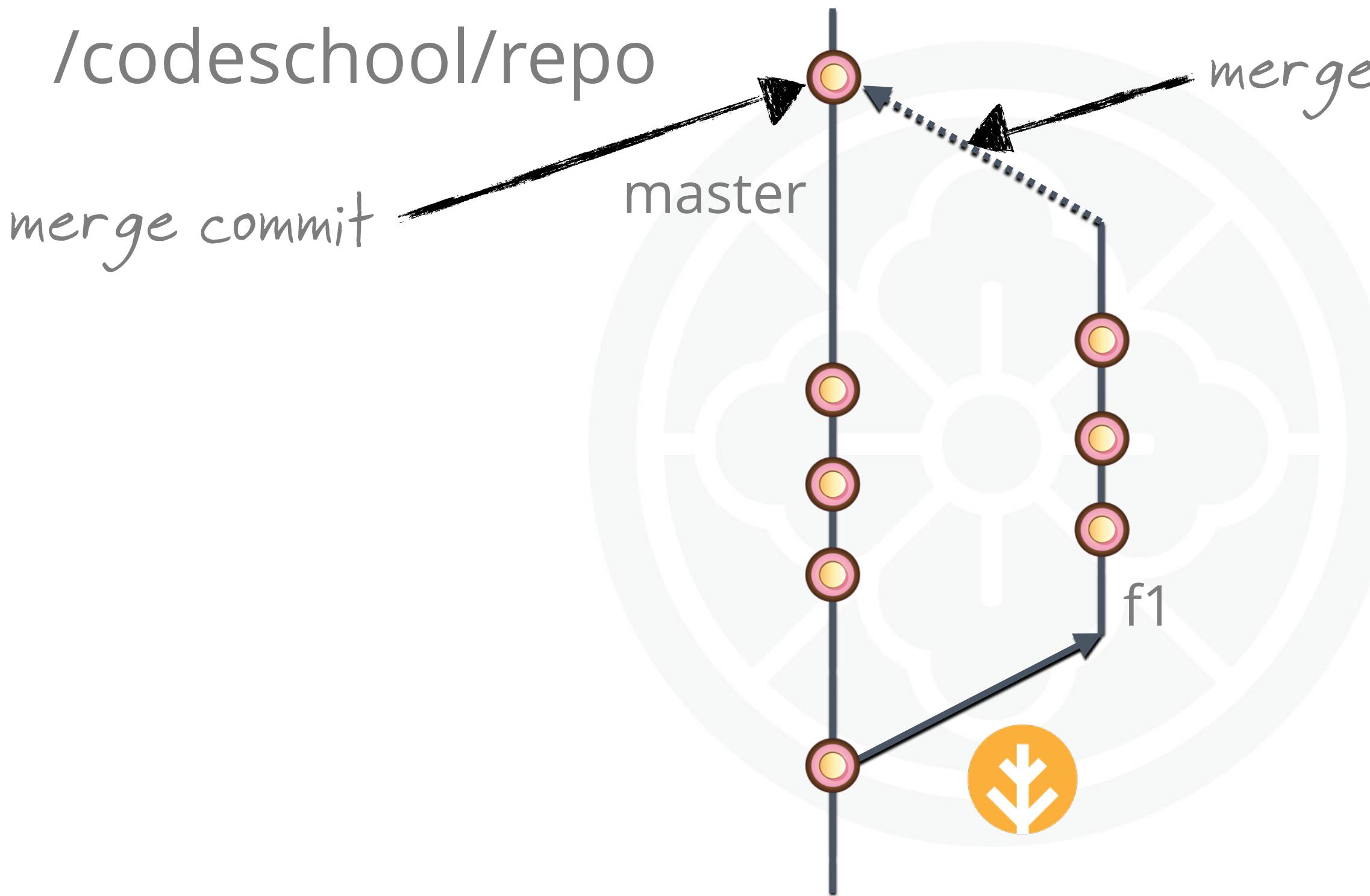
/codeschool/repo

Multiple developers can collaborate on each feature branch.

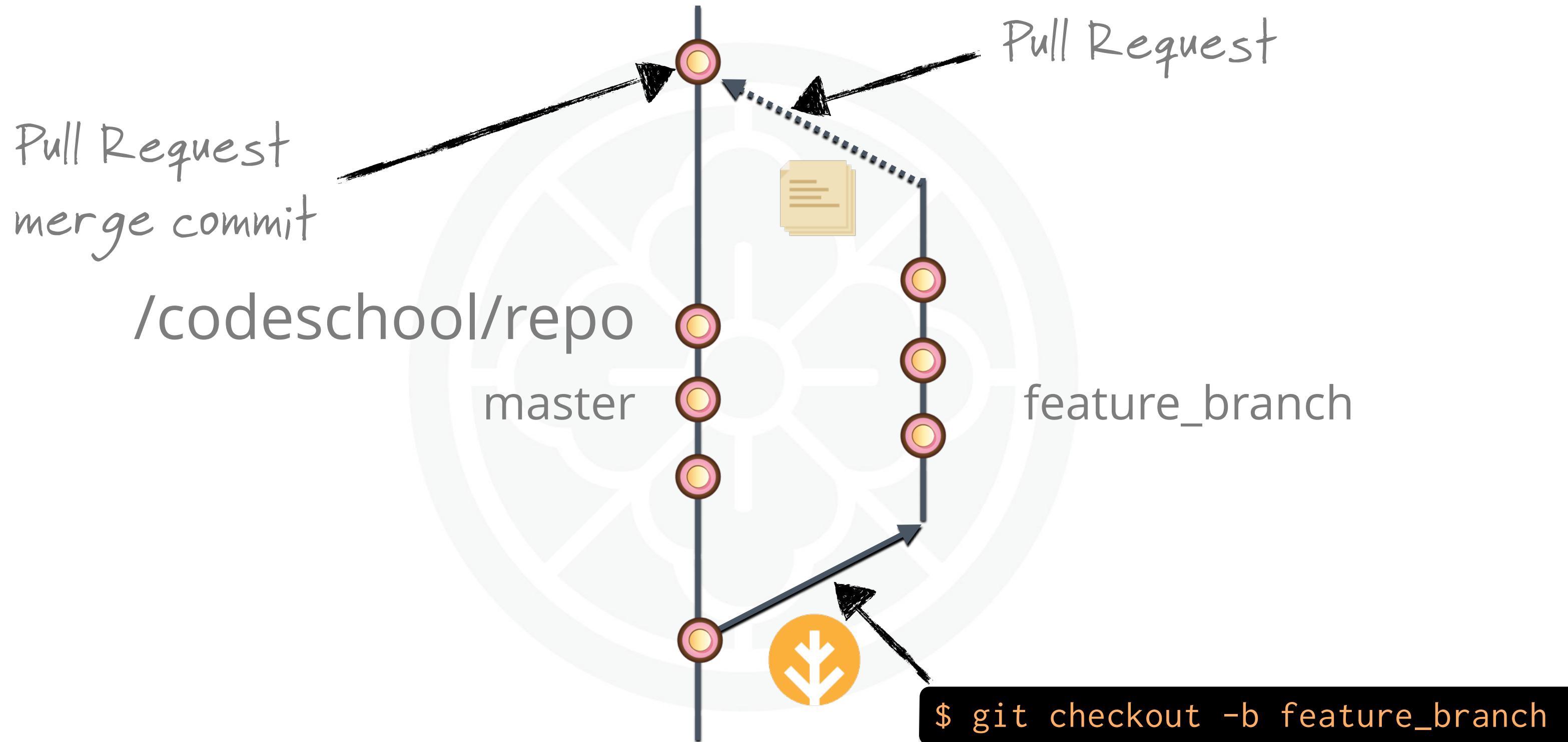
```
$ git fetch  
$ git branch -a  
$ git checkout f2
```



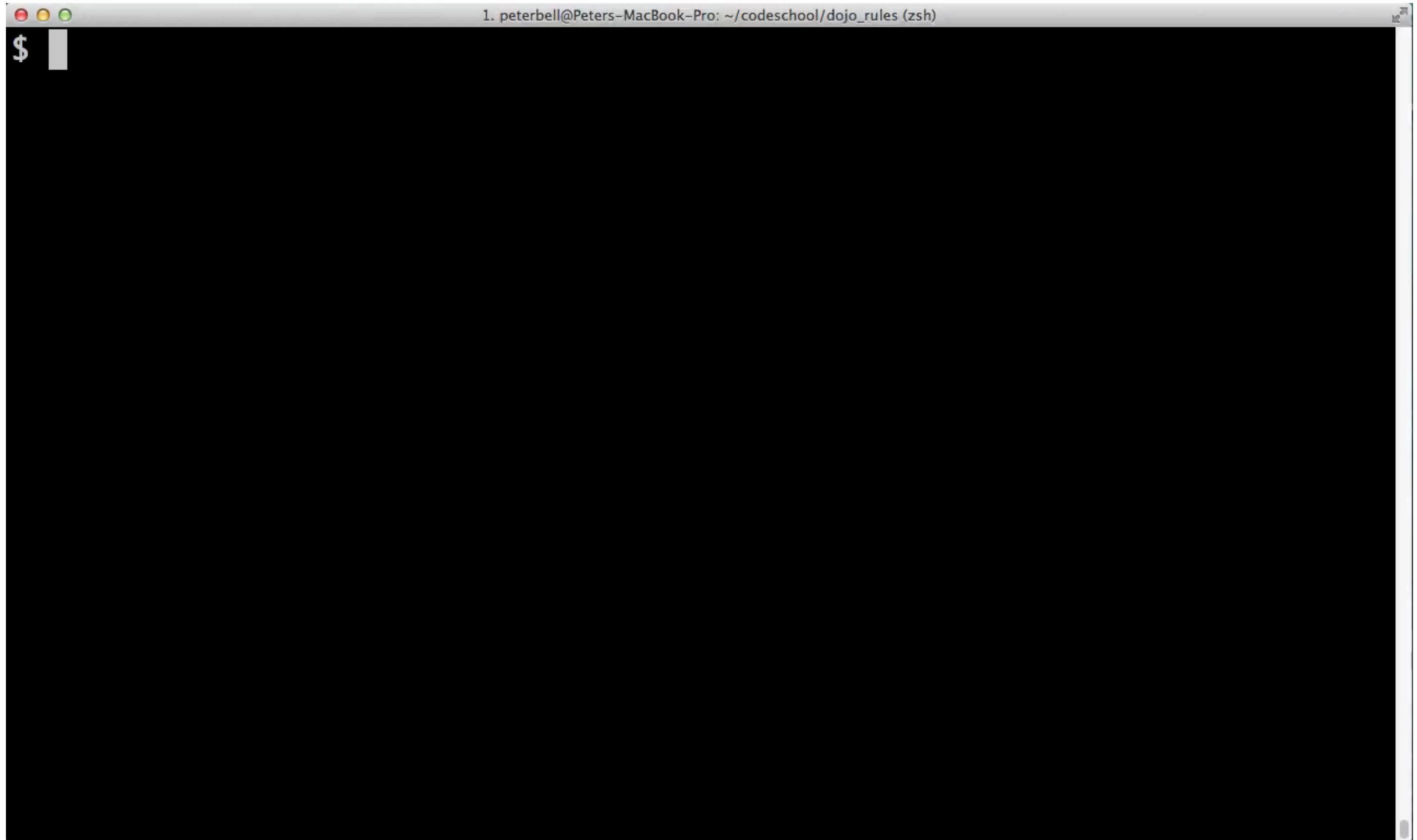
Merging Development Back into Master



Pull Requests for Reviewing Work



Submitting a Pull Request

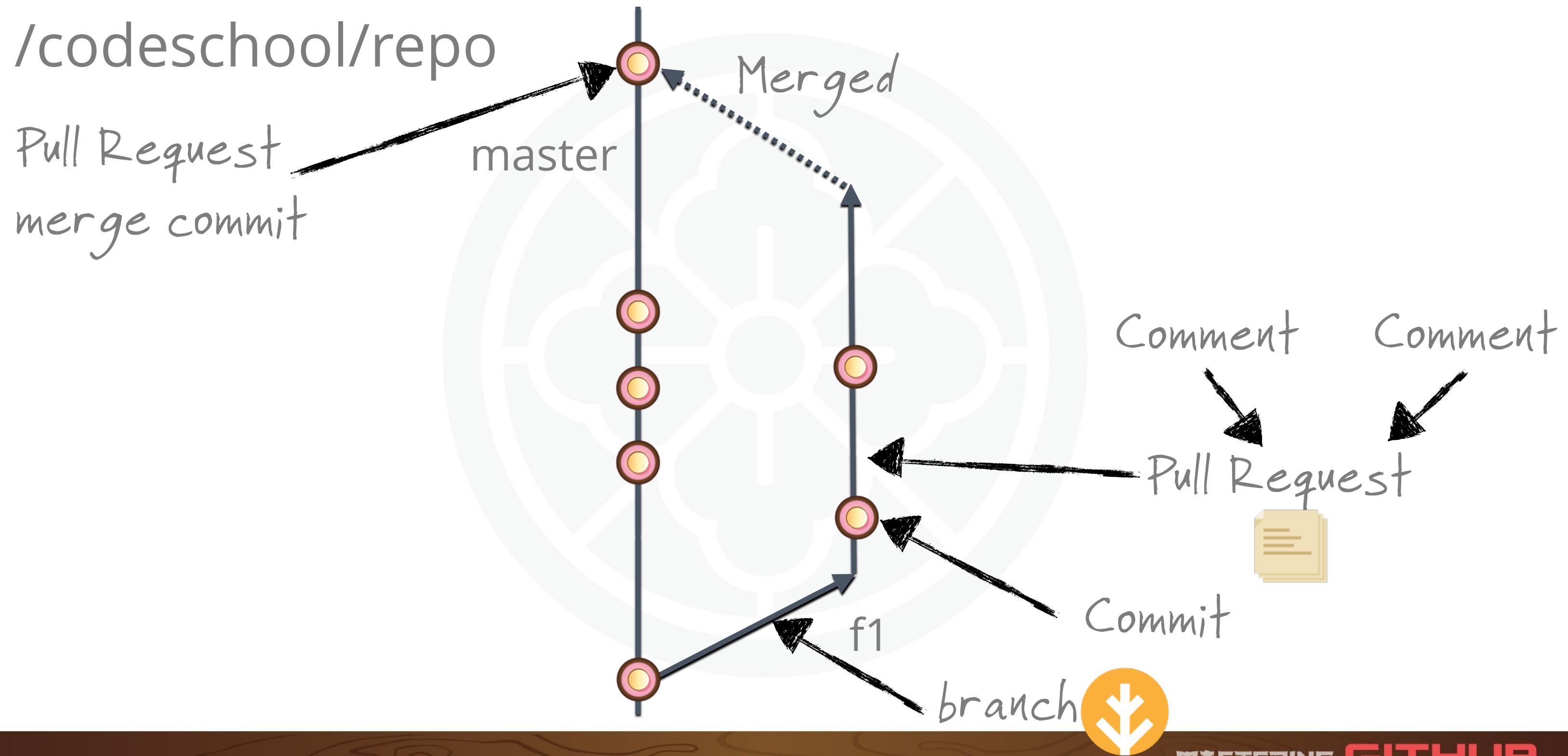


A screenshot of a Mac OS X terminal window. The window title bar reads "1. peterbell@Peters-MacBook-Pro: ~/codeschool/dojo_rules (zsh)". The main pane is black with a white cursor bar at the top. A single dollar sign (\$) character is visible on the left side of the cursor bar.

Collaborating on Pull Requests

Level 2: Section 2

Collaborating on Pull Requests



Types of Comments on Pull Requests

- on the PR
- on a commit
- on a line

Collaborating on Pull Requests

A screenshot of a GitHub pull request interface. The title bar shows the URL: https://github.com/deadlyvipers/dojo_rules/pull/3. The main content area displays a comment from user **PeterBell** (commented 26 minutes ago): "No description provided." Below the comment, there is a commit titled "Initial kill list" with hash **0f85e77**. A note says "Add more commits by pushing to the `kill_list` branch on `deadlyvipers/dojo_rules`". A green box highlights the message: "This pull request can be automatically merged. You can also merge branches on the command line." To the right, there are sections for Labels (None yet), Milestone (No milestone), Assignee (No one assigned), and Notifications (You're receiving notifications because you authored the thread). At the bottom, there is a comment input field with "Leave a comment" placeholder and a "Comment" button.

Initial kill list by PeterBell

[GitHub, Inc. \[US\] https://github.com/deadlyvipers/dojo_rules/pull/3](https://github.com/deadlyvipers/dojo_rules/pull/3)

Open PeterBell wants to merge 1 commit into `master` from `kill_list`

Conversation 0 Commits 1 Files changed 1

PeterBell commented 26 minutes ago

No description provided.

Initial kill list 0f85e77

Add more commits by pushing to the `kill_list` branch on `deadlyvipers/dojo_rules`.

This pull request can be automatically merged. You can also merge branches on the command line.

Write Preview Parsed as Markdown Edit in fullscreen

Leave a comment

Attach images by dragging & dropping, selecting them, or pasting from the clipboard.

ProTip Add comments to specific lines under Files changed.

Close Comment

Labels None yet

Milestone No milestone

Assignee No one assigned

Notifications You're receiving notifications because you authored the thread.

1 participant

Reviewing a Pull Request

Download all branches from GitHub.

```
$ git fetch
```

View all of the branches.

```
$ git branch -a
```

includes remote tracking branches in red

Checkout a local copy of a remote branch.

```
$ git checkout <branch_name>
```

Test code, make any changes and then commit and push changes.

```
$ <make edits>  
$ git commit  
$ git push
```

Reviewing a Pull Request

The screenshot shows a GitHub repository page for `deadlyvipers / dojo_rules`. The user `PeterBell` is logged in. There is one open pull request:

- All requests** (1)
- Open**
- Closed**
- Sort: Newest**
- New pull request**

The pull request details:

- First cut of equipment list** (#5)
- No description available
- by `githubteacher` 2 minutes ago
- `equipment_list`

On the right side, there is a vertical toolbar with various icons for repository management.

© 2014 GitHub, Inc. Terms Privacy Security Contact

Best Practices for Merging Pull Requests

Level 2: Section 3

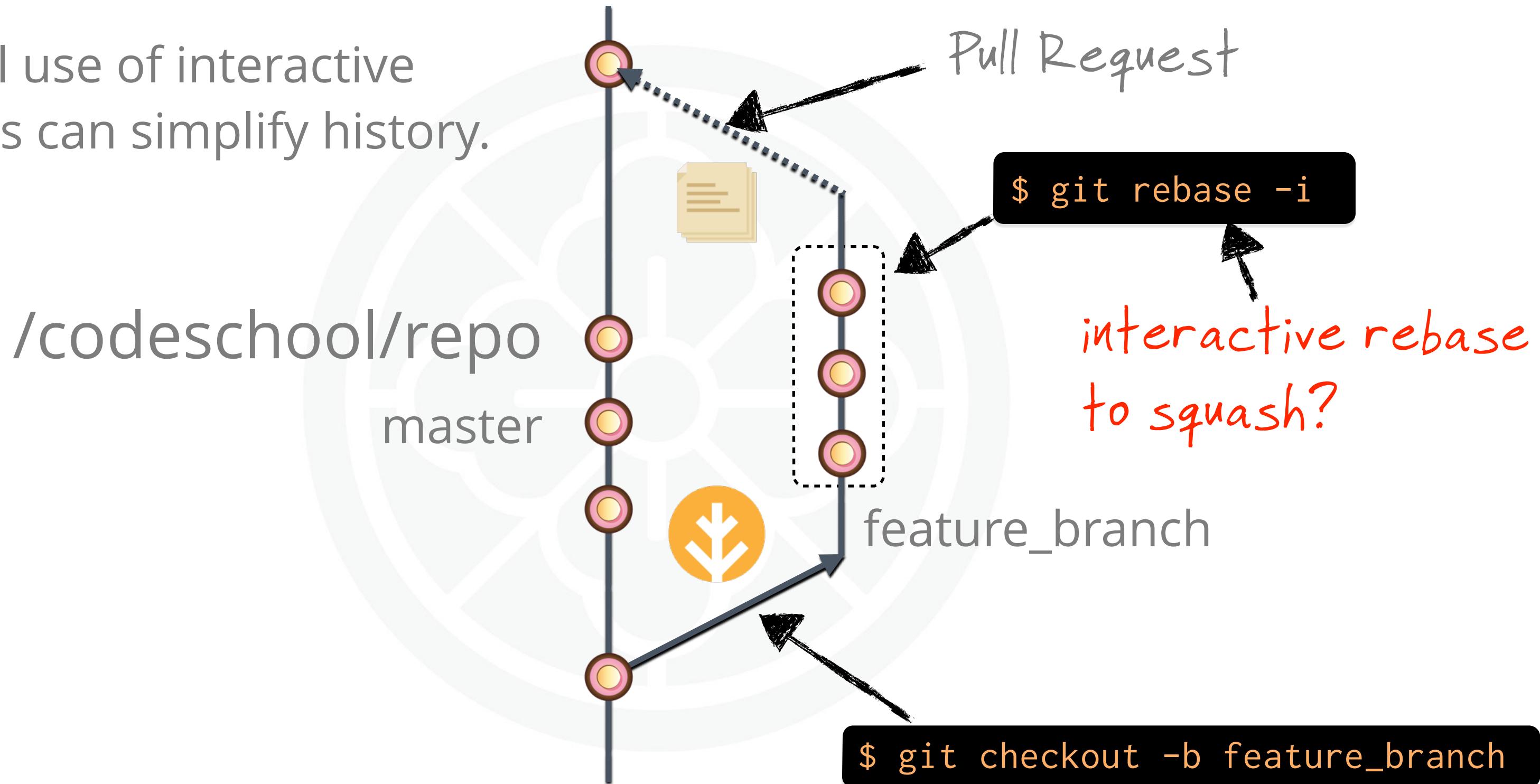
Considerations when Merging Pull Requests

There are three key considerations - should you:

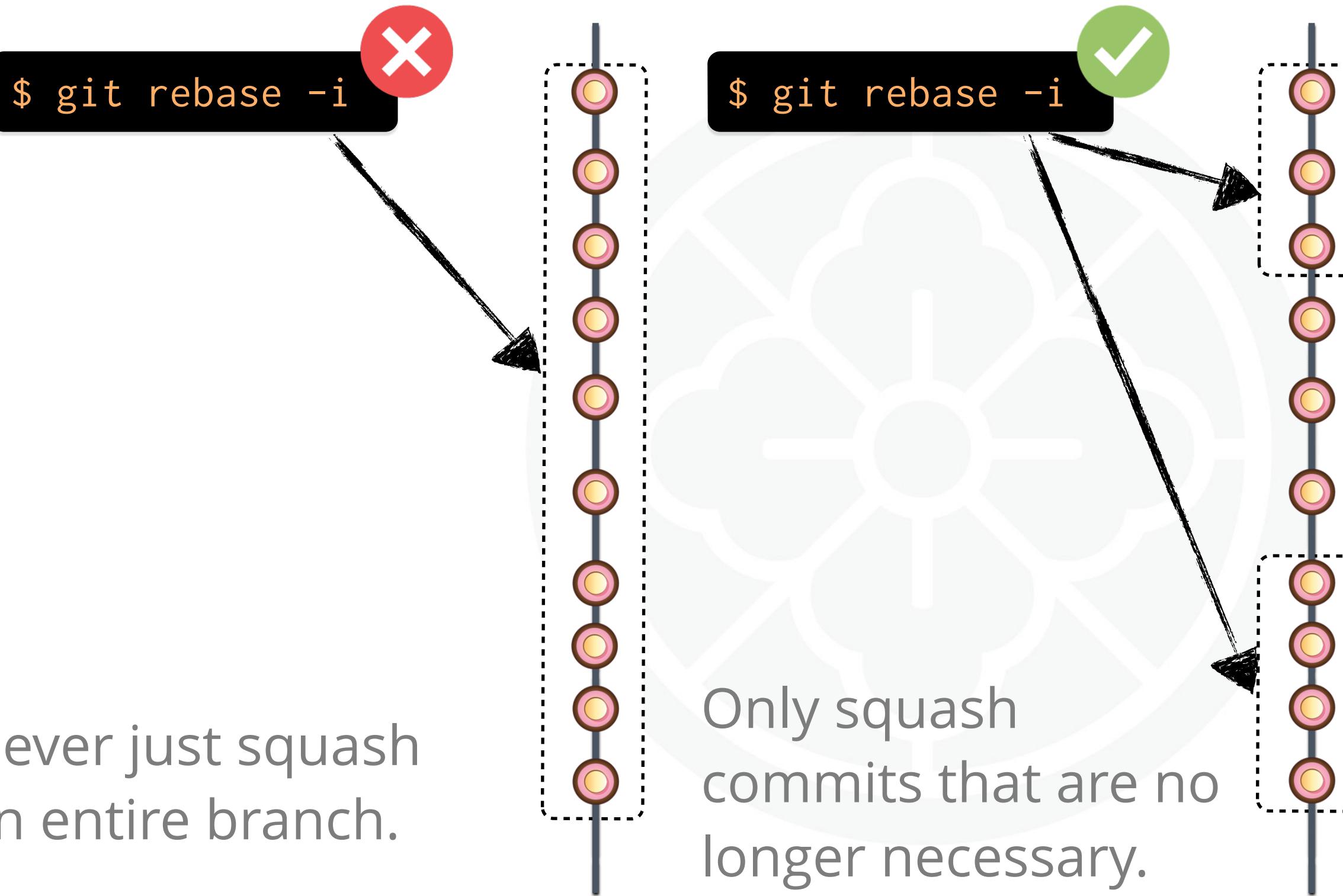
- Interactively rebase to simplify history?
- Rebase to create a linear history?
- Use a fast-forward or recursive merge?

Interactive Rebases to Simplify History?

Careful use of interactive rebases can simplify history.



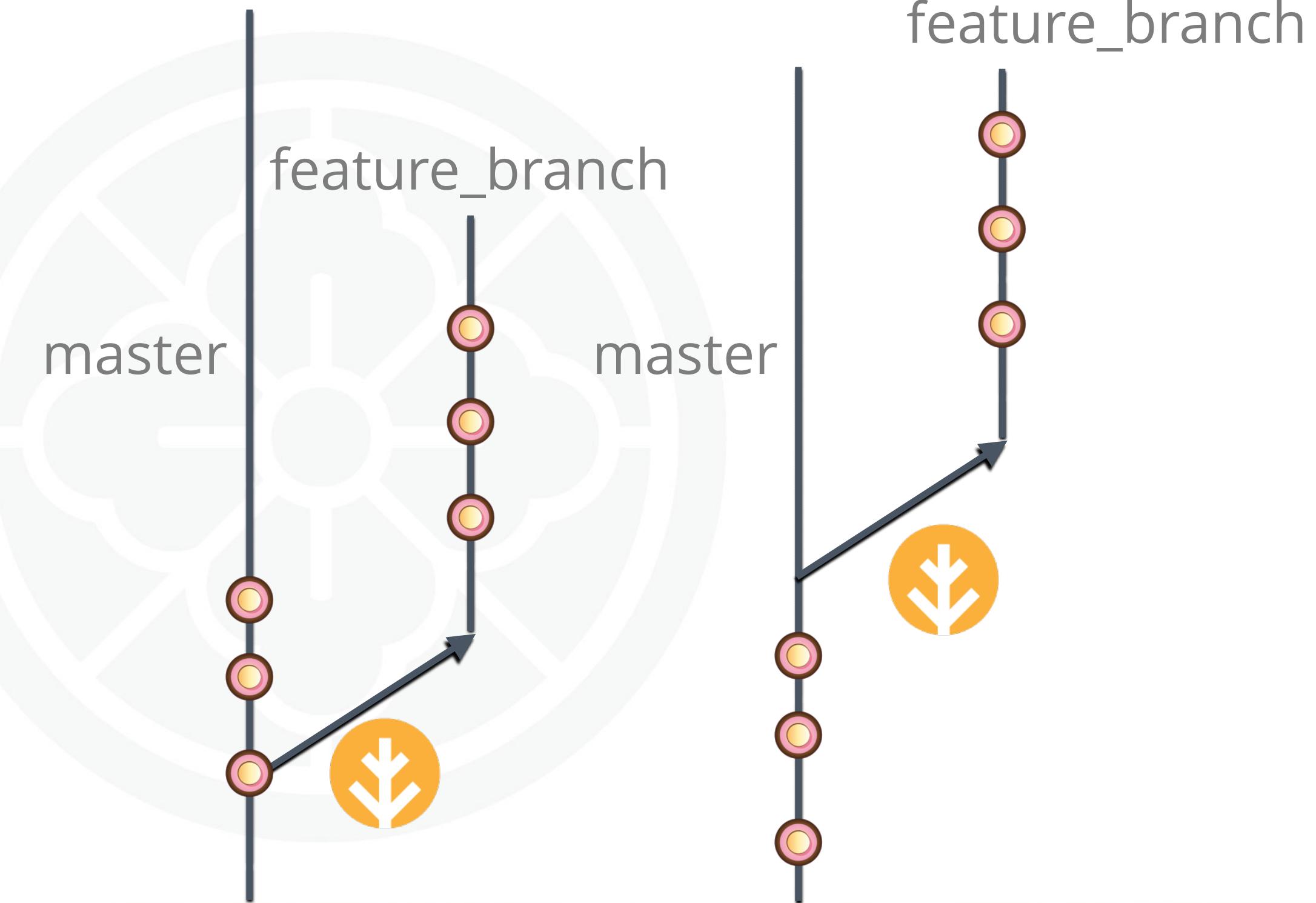
Interactive Rebases to Simplify History?



Rebasing to Create a Linear History?

rebase

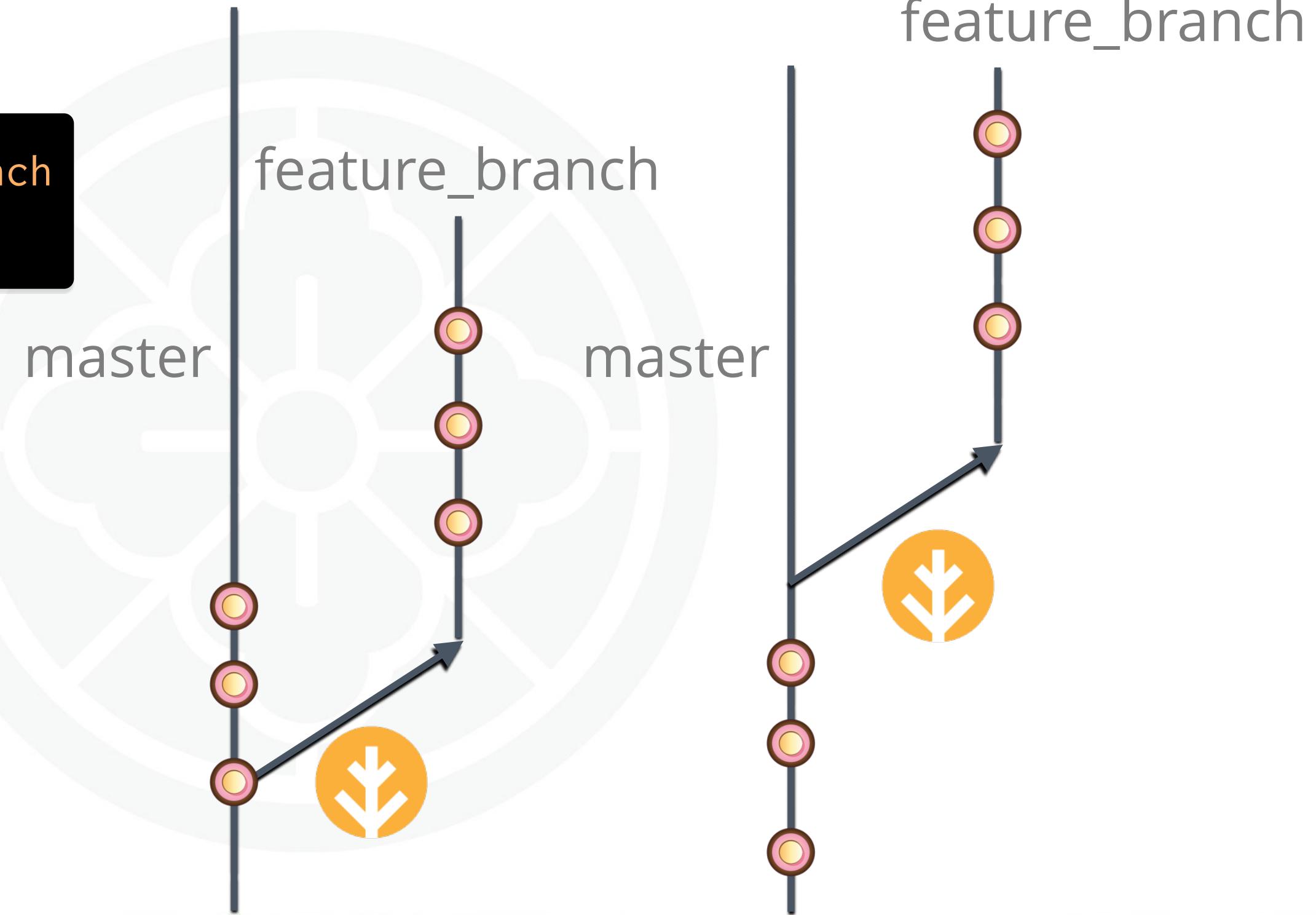
- Creates simpler history 
- Makes it look like features delivered  one at a time
- Changes commit hashes so your team needs to know what they're doing 



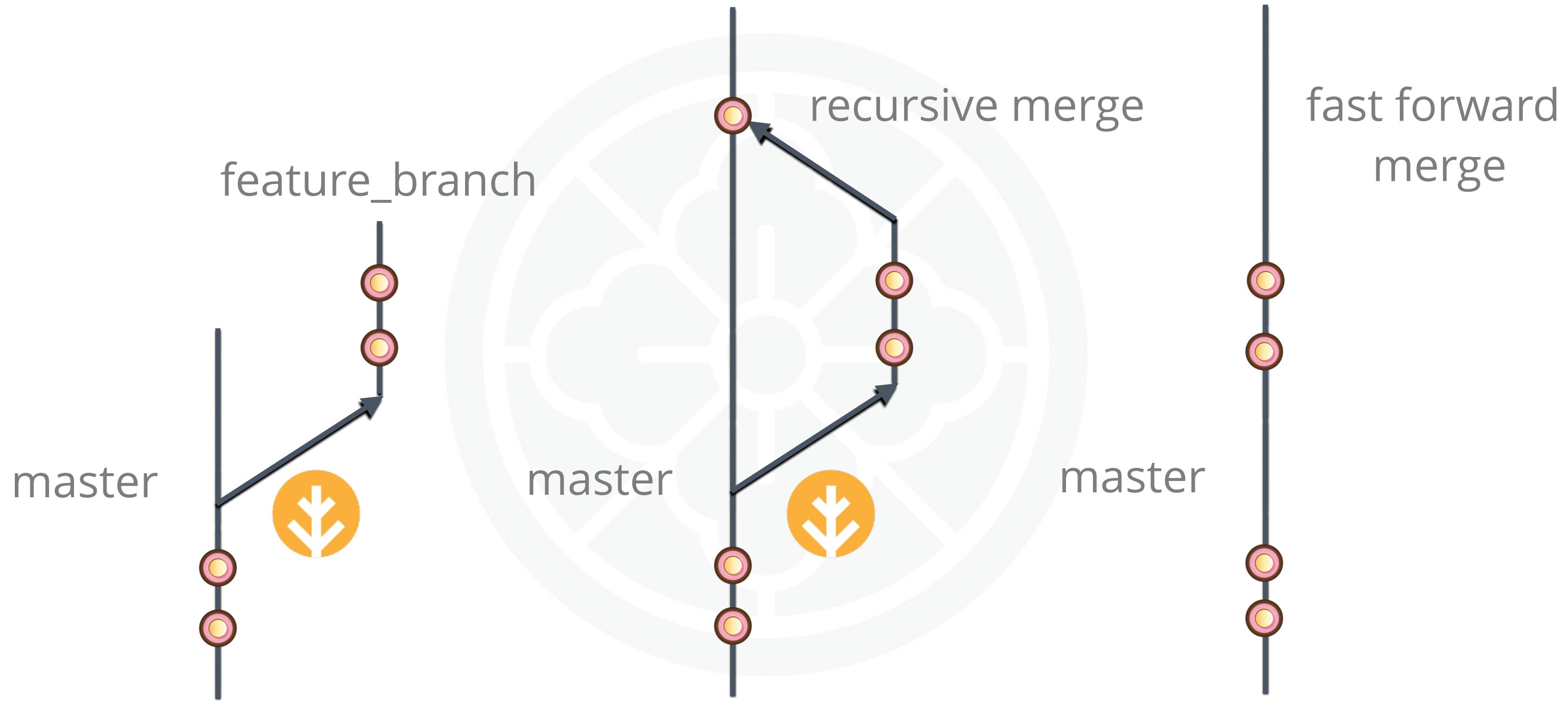
Rebasing to Create a Linear History?

rebase

```
$ git checkout feature_branch  
$ git rebase master
```

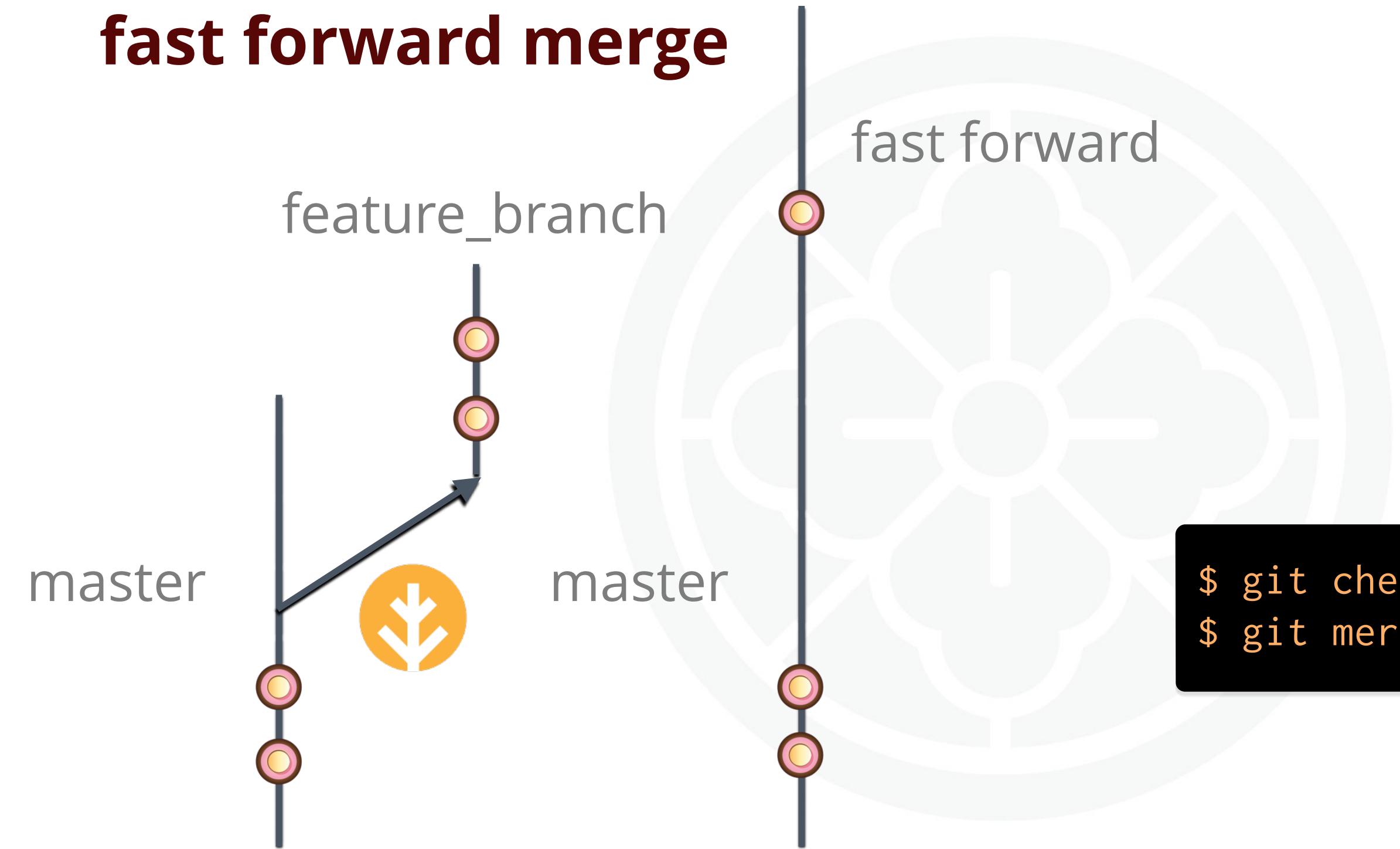


Fast Forward or Recursive Merges?



Fast Forward or Recursive Merges?

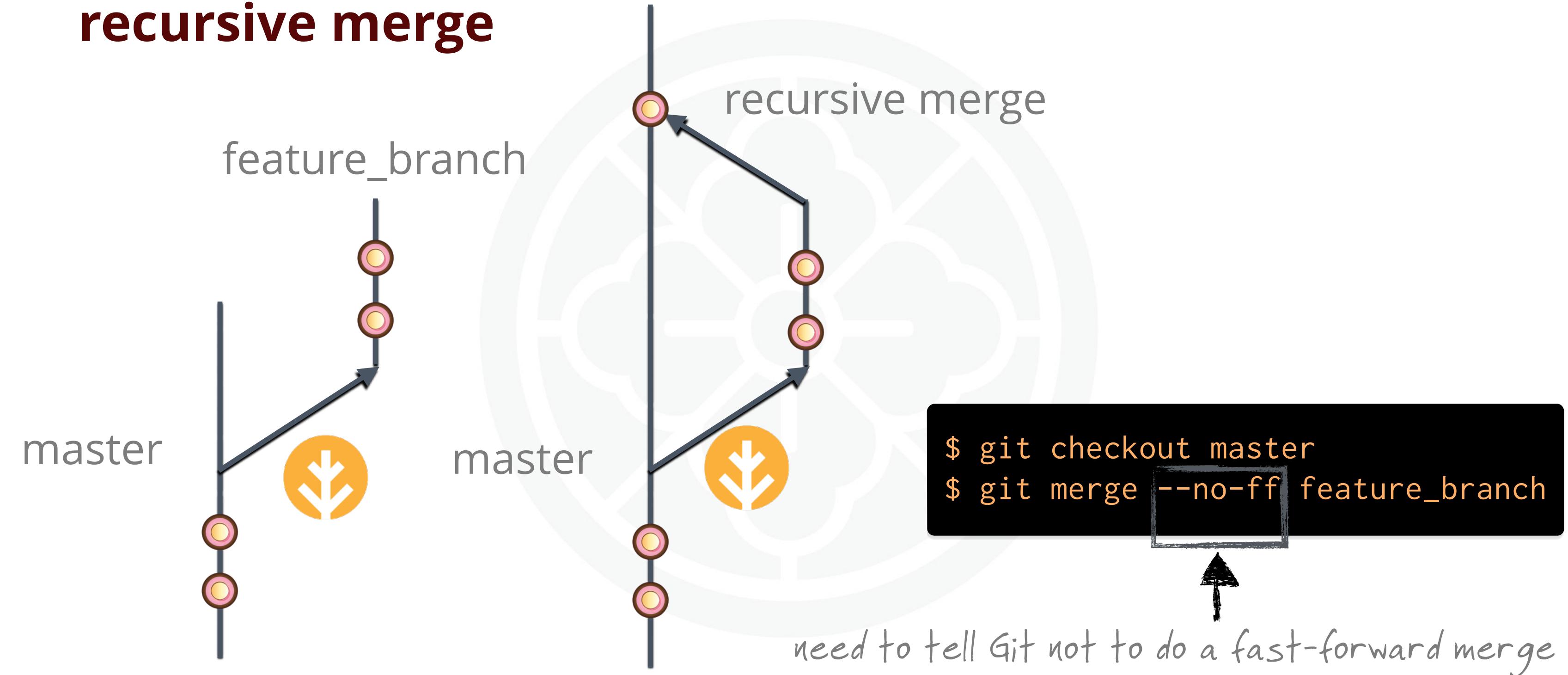
fast forward merge



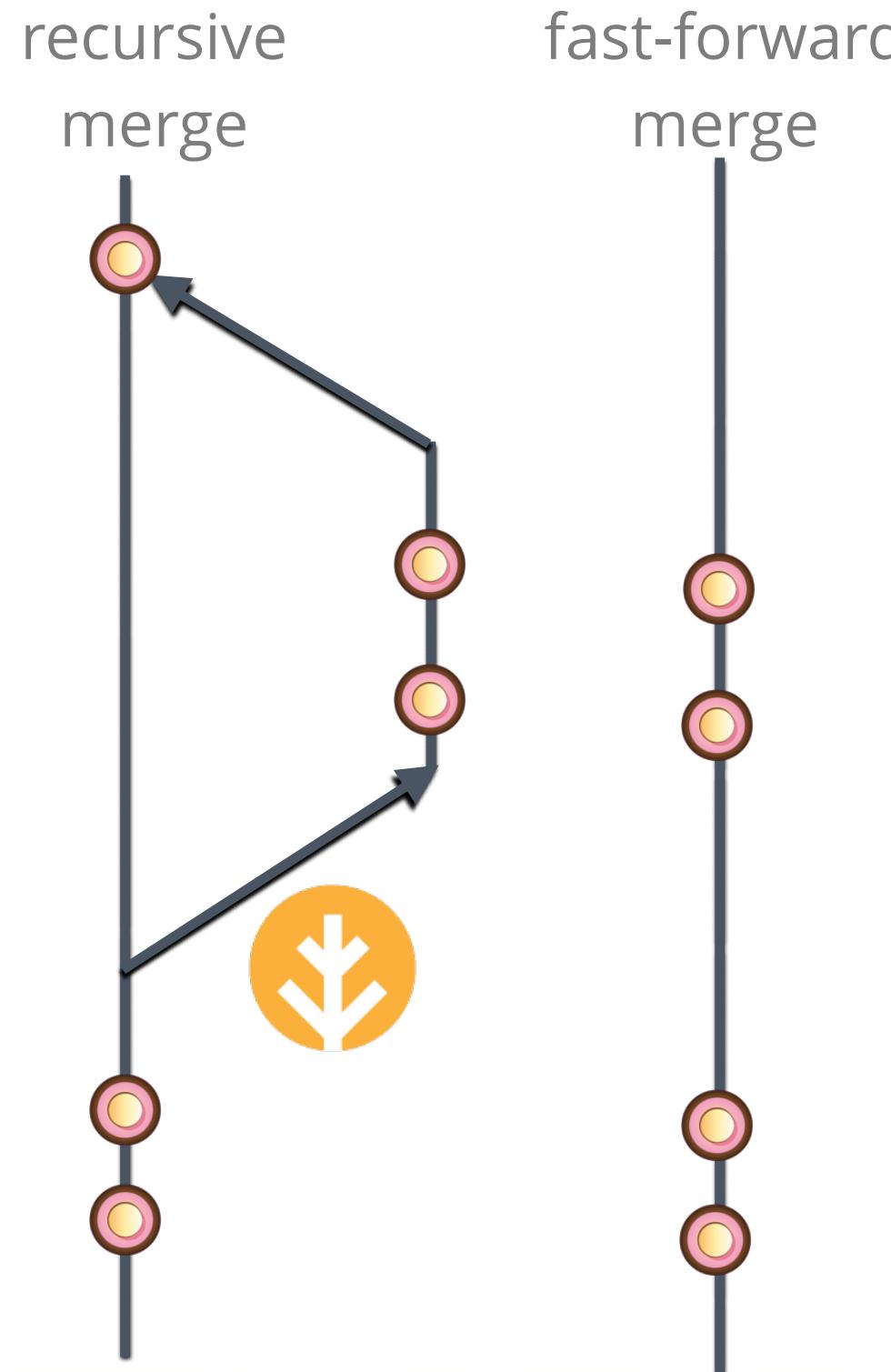
```
$ git checkout master  
$ git merge feature_branch
```

Fast Forward or Recursive Merges?

recursive merge



Fast Forward or Recursive Merges?



Fast Forward:

- Less commits

Recursive:

- Clear what was done on a branch
- Single commit to revert

Considerations when Merging Pull Requests

There are three key considerations - should you:

Interactively rebase to simplify history?

Only when it leaves enough context.

Rebase to create a linear history?

If your team is comfortable with it.

Fast forward or recursive merges?

Recursive provides more clarity.

Considerations when Merging Pull Requests

The screenshot shows a GitHub pull request interface for a repository named `deadlyvipers/dojo_rules`. The pull request is titled "Creating an action plan #7" and is currently open. The summary indicates that **PeterBell** wants to merge 5 commits from the `action_plan` branch into the `master` branch.

Key details visible in the interface:

- Commits:** 5 commits from **PeterBell** (The Bride) have been added to the pull request.
- Files changed:** 1 file has been changed.
- Owner:** **PeterBell**
- Labels:** None yet
- Milestone:** No milestone
- Assignee:** No one assigned
- Notifications:** You're receiving notifications because you authored the thread.
- Automated Merge:** A message states "This pull request can be automatically merged." and provides a link to "Merge pull request".
- Branch Status:** The `action_plan` branch is ahead by 6 commits and behind by 0 commits.

The commit history from **PeterBell** (The Bride) includes the following commits:

Commit Message	SHA
Gotta kill Bill	49fa261
Need help from Hattori	0179457
Kill Vernita and her daughter first	aa1079f
Kill Bill first - then the others	13ef905
Leave Bill until last	2e032e5

At the bottom, there are options to "Write" or "Preview" the pull request, and links to "Parsed as Markdown" and "Edit in fullscreen".



• MASTERING • GITHUB

• MASTERING • GITHUB

Tags

Level 3: Section 1

When to Use Tags



Tag every time you push to production

Unless *every* commit on master goes to production

Tags are documentation. Use them to keep track of all production releases.

The Three Different Types of Tags



- **Lightweight** - just a tag, no message or tagger
- **Signed** - uses public key to prove identity of tagger
- **Annotated** - adds info on who tagged, when and why 

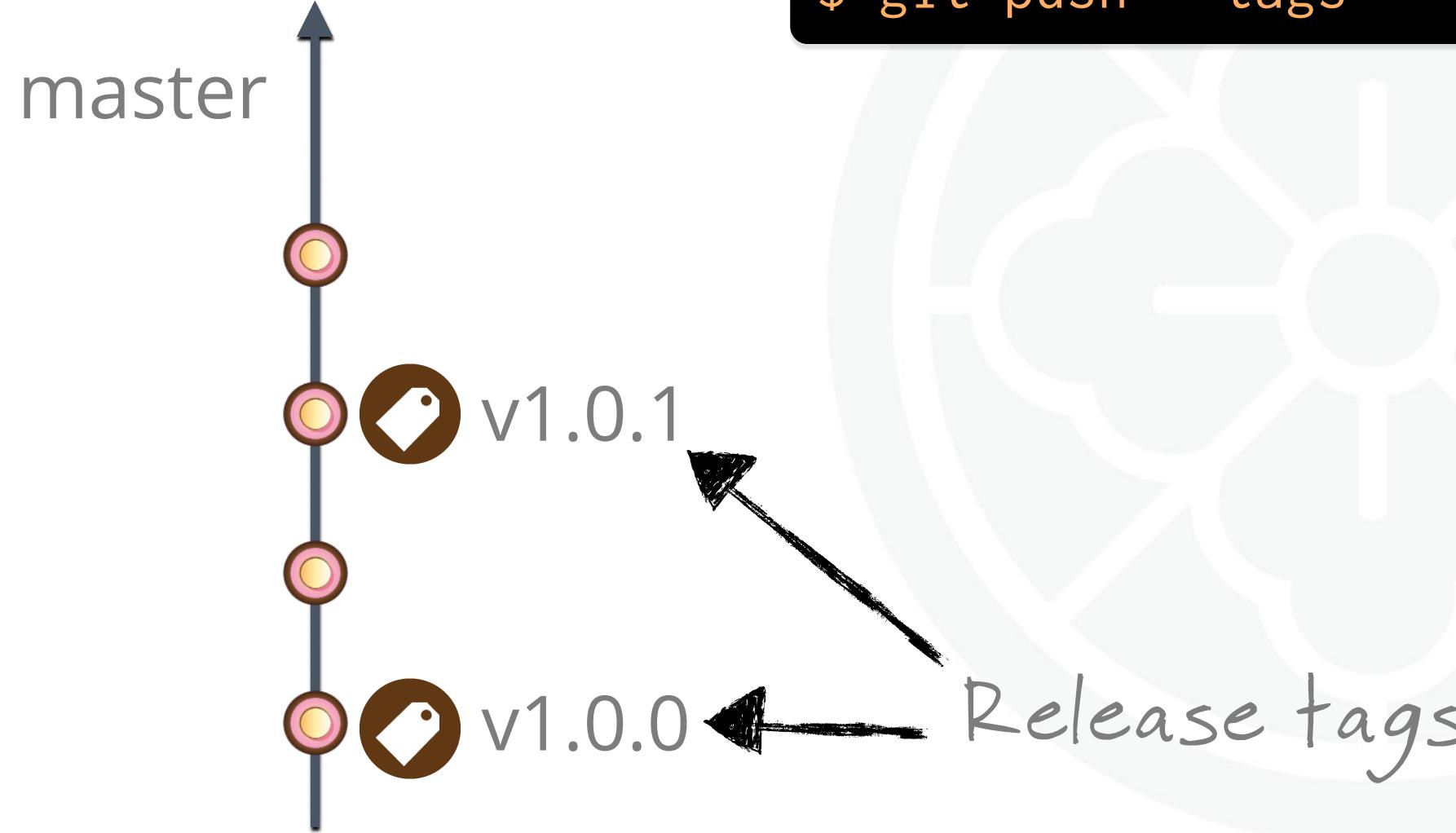
```
$ git tag  
$ git tag -s  
$ git tag -a
```

Using Annotated Tags

```
$ git tag -a v1.3.2 -m "Tag message"
```

```
$ git push --tags
```

push changes and new tags



Best Practices for Naming Tags

```
$ git tag -a v1.3.2 -m "Tag message"
```

Tagging suggestions

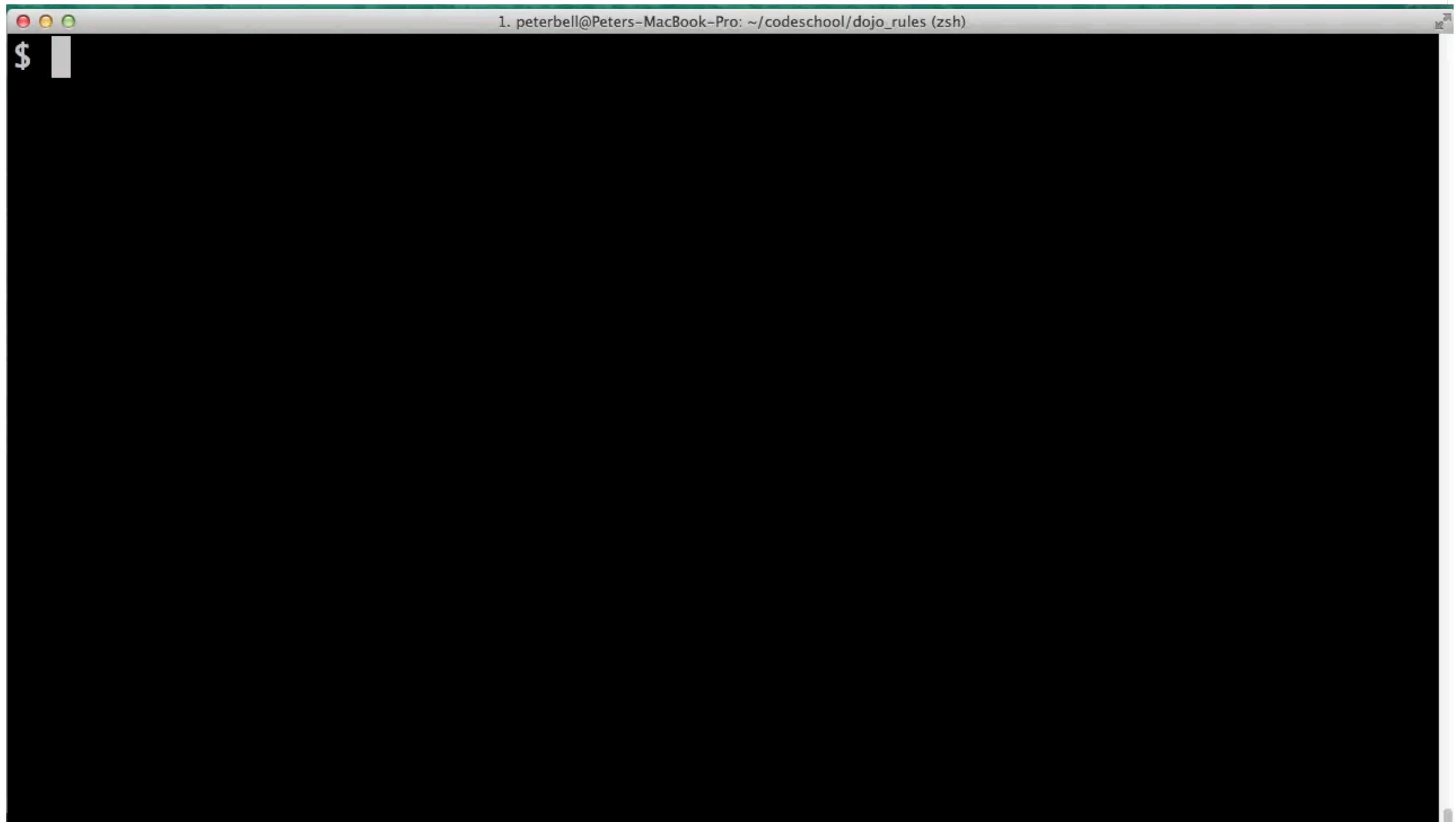
It's common practice to prefix your version names with the letter v.

Some good tag names might be v1.0 or v2.3.4.

If the tag isn't meant for production use, add a pre-release version after the version name. Some good pre-release versions might be v0.2-alpha or v5.9-beta.3.

Semantic versioning is a best practice
Release: major.minor.patch

Tagging a release



A screenshot of a Mac OS X terminal window. The window title bar reads "1. peterbell@Peters-MacBook-Pro: ~/codeschool/dojo_rules (zsh)". The terminal prompt shows a dollar sign (\$) followed by a vertical white bar indicating where text is being typed. The rest of the screen is blank black space.

Release Branches

Level 3: Section 2

Introducing Release Branches

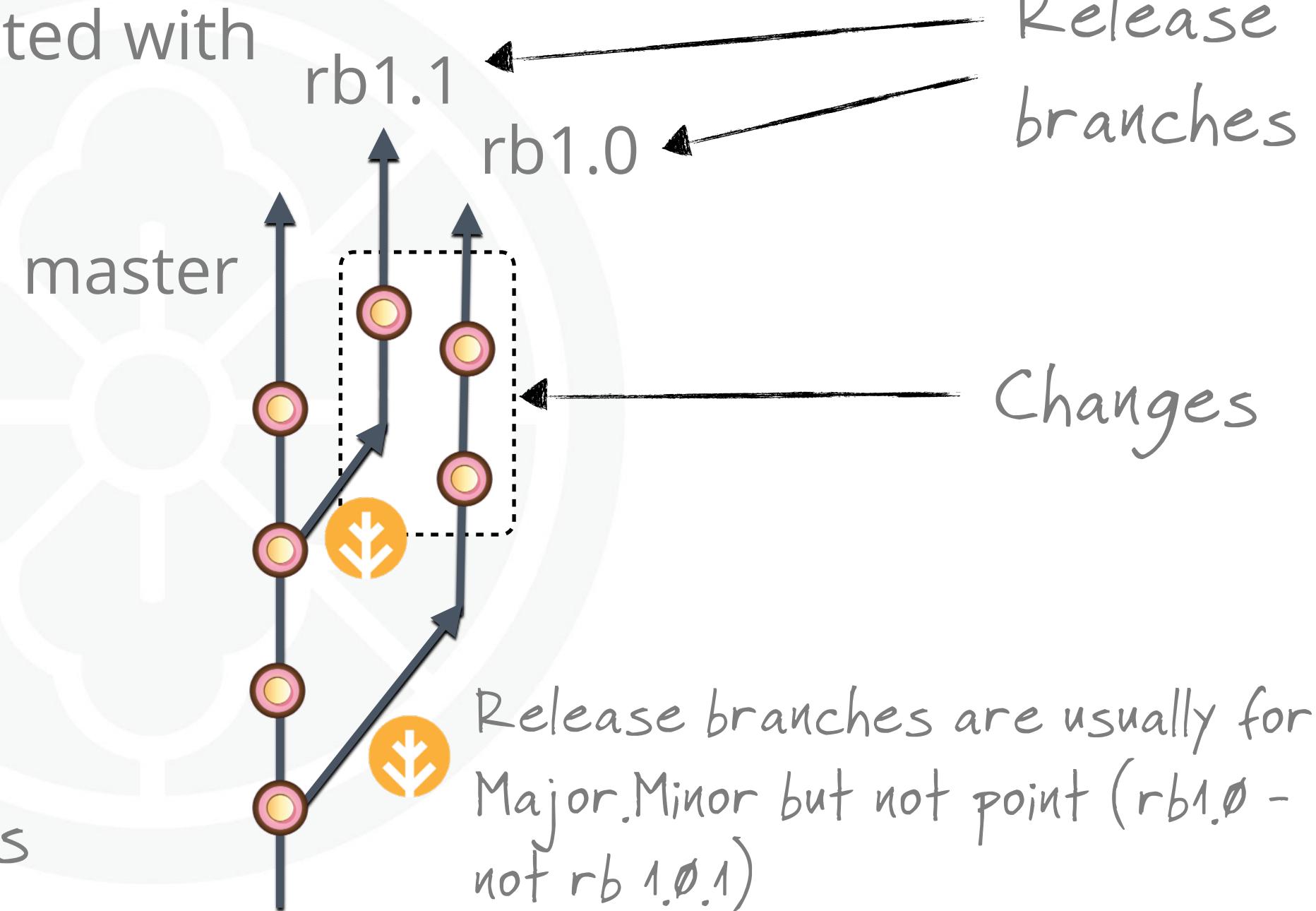
Generally you only need tags for releases, but what if you need to make a change to a production release without affecting master?

Then you need a release branch!

Release Tags vs. Release Branches

Release tags point to a single commit.

Release branches can be updated with new commits

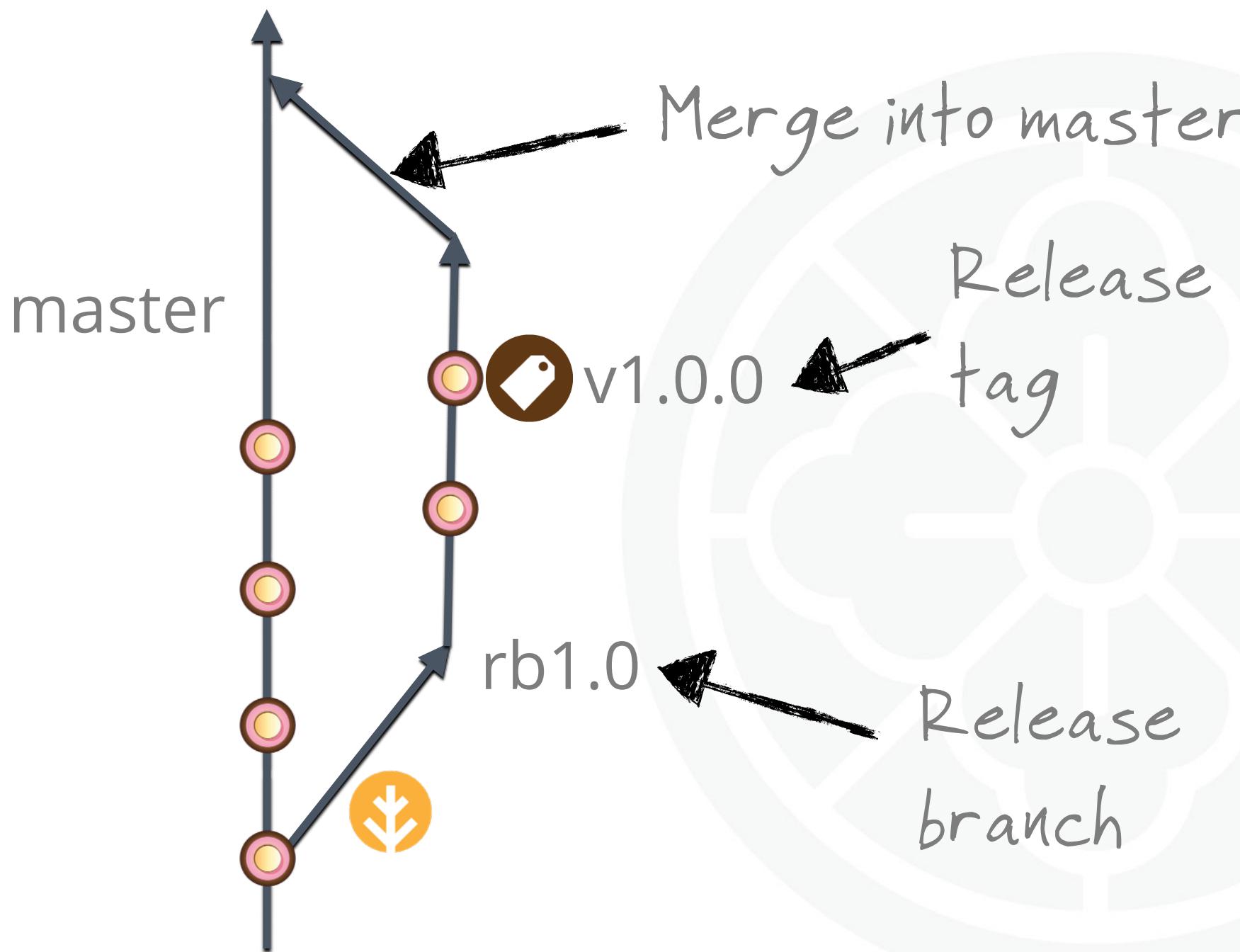


Use Cases for Release Branches

There are three key reasons for creating release branches:

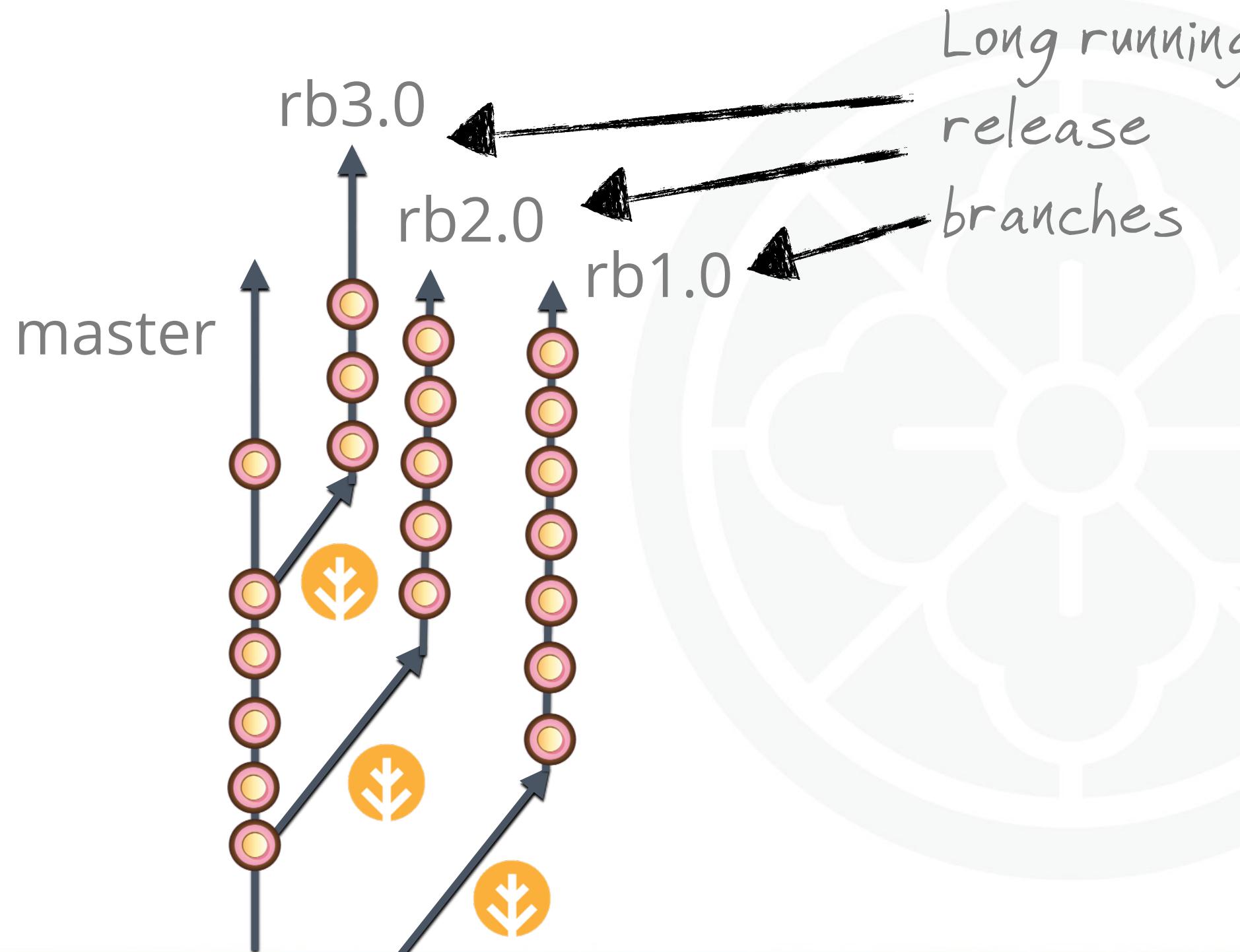
- Manual QA
- Long running releases
- On demand hot fixes

Release Branches - Manual QA



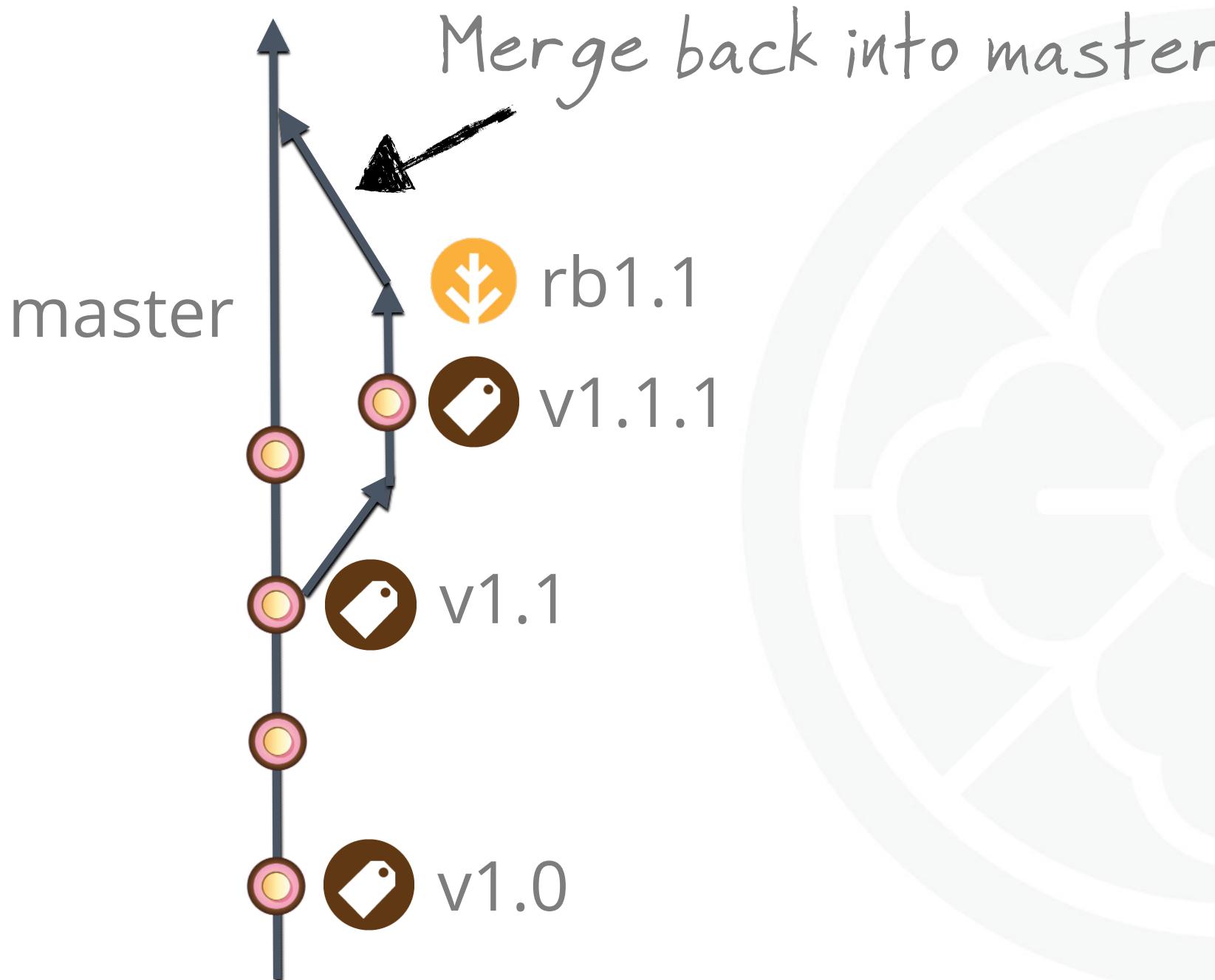
If you know you're going to have hot fixes from your manual QA process, create a release branch upfront.

Release Branches - Long Running Releases



If you need to support long running releases, you'll need to keep a branch for each one.

Release Branches - On Demand Hotfixes



```
$ git checkout v1.1  
$ git checkout -b rb1.1  
<hotfix>  
$ git commit -m "Hotfix"  
$ git tag v1.1.1 -m "Hotfix"  
$ git checkout master  
$ git merge rb1.1 -m "Merged hotfix"  
$ git branch -d rb1.1
```

You can create release branches on the fly from release tags - no need to create upfront

Releases in GitHub

Level 3: Section 3

Releases

If you want to share downloadable binaries plus additional notes with each of your tags, use the "releases" feature in GitHub.

Benefits of releases:

- Host binary downloads without checking into Git
- Provide additional documentation/release notes

Releases

◀ Releases

Tag version @ Target: master ▾

Choose an existing tag, or create a new tag on publish

Release title

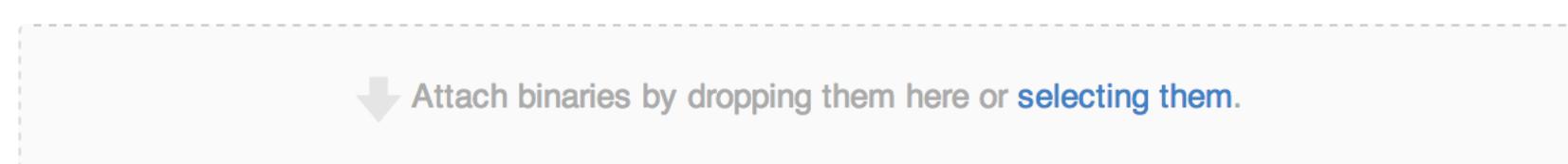
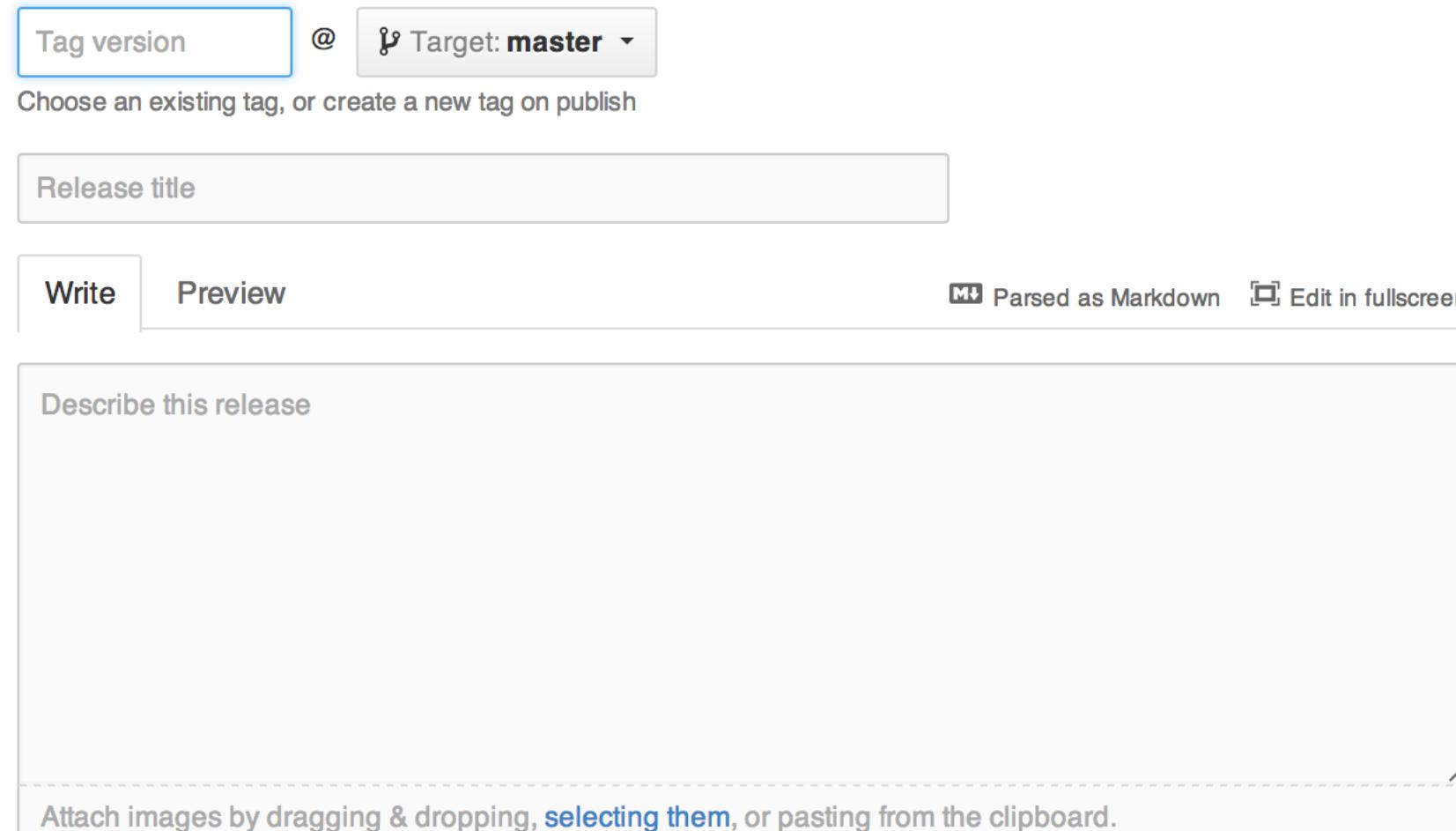
Write Preview Parsed as Markdown Edit in fullscreen

Describe this release

Attach images by dragging & dropping, [selecting them](#), or pasting from the clipboard.

↓ Attach binaries by dropping them here or [selecting them](#).

This is a pre-release
We'll point out that this release is identified as non-production ready.



Releases:

- Tag - create in Git or on GitHub
- Binaries - upload without adding to repo
- Notes - more comprehensive release notes

Releases

The screenshot shows a GitHub release page for the repository `deadlyvipers / dojo_rules`. The release is titled `v1.1.0`, which was tagged 2 minutes ago. It includes 1 commit to the master branch since the tag. The release notes mention an "action plan". Two download links are provided: "Source code (zip)" and "Source code (tar.gz)". The GitHub interface includes a sidebar with navigation links like Explore, Gist, Blog, Help, and a user profile for PeterBell.

Release v1.1.0 · deadlyvipers / dojo_rules · Semantic Versioning 2.0.0

GitHub, Inc. [US] https://github.com/deadlyvipers/dojo_rules/releases/tag/v1.1.0

This repository Search or type a command Explore Gist Blog Help PeterBell

PUBLIC deadlyvipers / dojo_rules Unwatch 1 Star 0 Fork 2

< Releases Edit tag Delete

v1.1.0
tagged 2 minutes ago · 1 commit to master since this tag

v1.1.0 - with action plan

Source code (zip) Source code (tar.gz)

© 2014 GitHub, Inc. Terms Privacy Security Contact Status API Training Shop Blog About



• MASTERING • GITHUB

• MASTERING • GITHUB

Issues

Level 4: Section 1

Why Use Issues?

Issues can be used for:

- Tracking bugs
- Managing features

Enabling Issues for a Project

Go to: settings/options

Features

Wikis

GitHub Wikis are the simplest way to let others contribute content. Any GitHub user can create and edit pages to use for documentation, examples, support, or anything you wish.

Restrict editing to collaborators only

Public wikis will still be readable by everyone.

Issues

GitHub Issues adds lightweight issue tracking tightly integrated with your repository. Add issues to milestones, label issues, and close & reference issues from commit messages.



Check box to implement issues.

Accessing Issues

- **public repo** - anyone can access
- **private repo** - only collaborators

Listing Issues

Click on “issues” in right hand nav

The screenshot shows a list of three open issues on GitHub:

- Difficult to use without a keyboard or mouse** (#3) - Opened by PeterBell 21 minutes ago. 1 comment.
- Can't view the images on my flip phone** (#2) - Opened by PeterBell 21 minutes ago.
- Doesn't work on IE3** (#1) - Opened by PeterBell 22 minutes ago.

At the top of the page, there is a green "New Issue" button. Below it, there are filters for "3 Open" and "0 Closed" issues, and a sorting dropdown set to "Sort: Newest". On the far right, there is a sidebar with various navigation links, one of which is labeled "Issues" with a circled exclamation mark icon. A black arrow points from the text "Click on ‘issues’ in right hand nav" to the "Issues" link in the sidebar.

Filtering Issues

use labels like "bugs"
and "features"

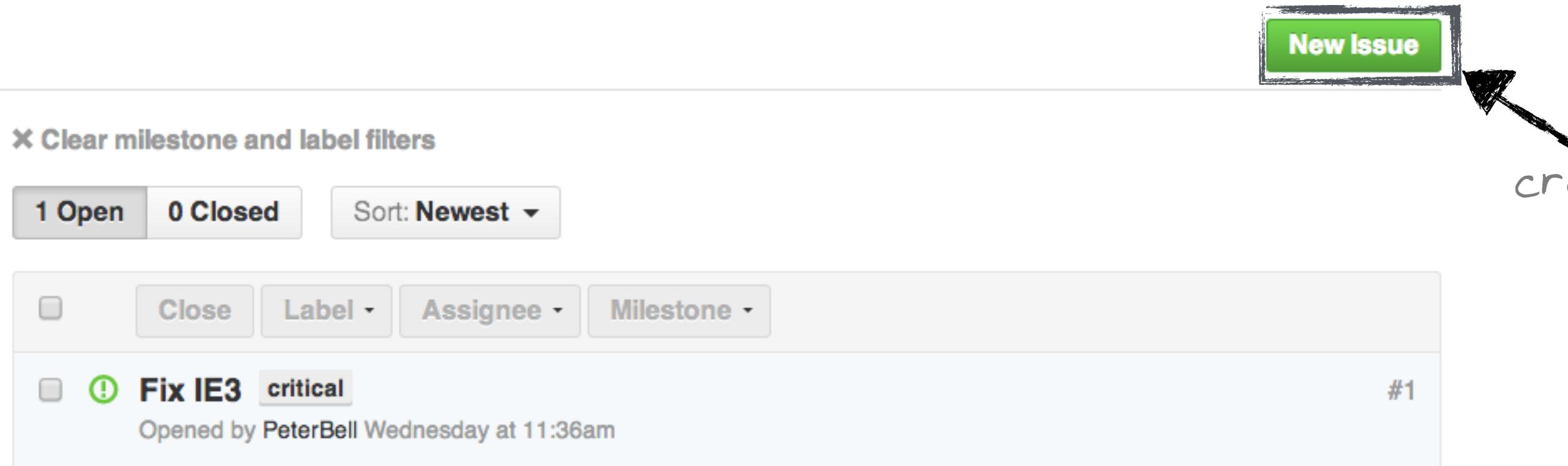
The screenshot shows the GitHub 'Everyone's Issues' page. At the top, there are filters for 'Assign to you' (0), 'Created by you' (1), and 'Mentioning you' (0). Below these, a section for 'No milestone selected' includes a gear icon. A large arrow points from the handwritten note about labels to the 'Labels' section. The 'Labels' section is highlighted with a black border and contains a list of labels: 'critical' (selected, with an 'X' icon to its right), 'bug' (with an orange exclamation mark icon and a count of 0), 'duplicate' (with a grey exclamation mark icon and a count of 0), 'enhancement' (with a blue plus icon and a count of 0), 'help wanted' (with a green question mark icon and a count of 0), 'invalid' (with a grey exclamation mark icon and a count of 0), and 'question' (with a pink question mark icon and a count of 0).

The screenshot shows a modal window titled 'Clear milestone and label filters'. It displays '1 Open' issues and '0 Closed' issues, sorted 'Newest'. The modal includes buttons for 'Close', 'Label', 'Assignee', and 'Milestone'. Below these buttons, a specific issue is listed: 'Fix IE3 critical' (opened by PeterBell on Wednesday at 11:36am).

Keyboard shortcuts available

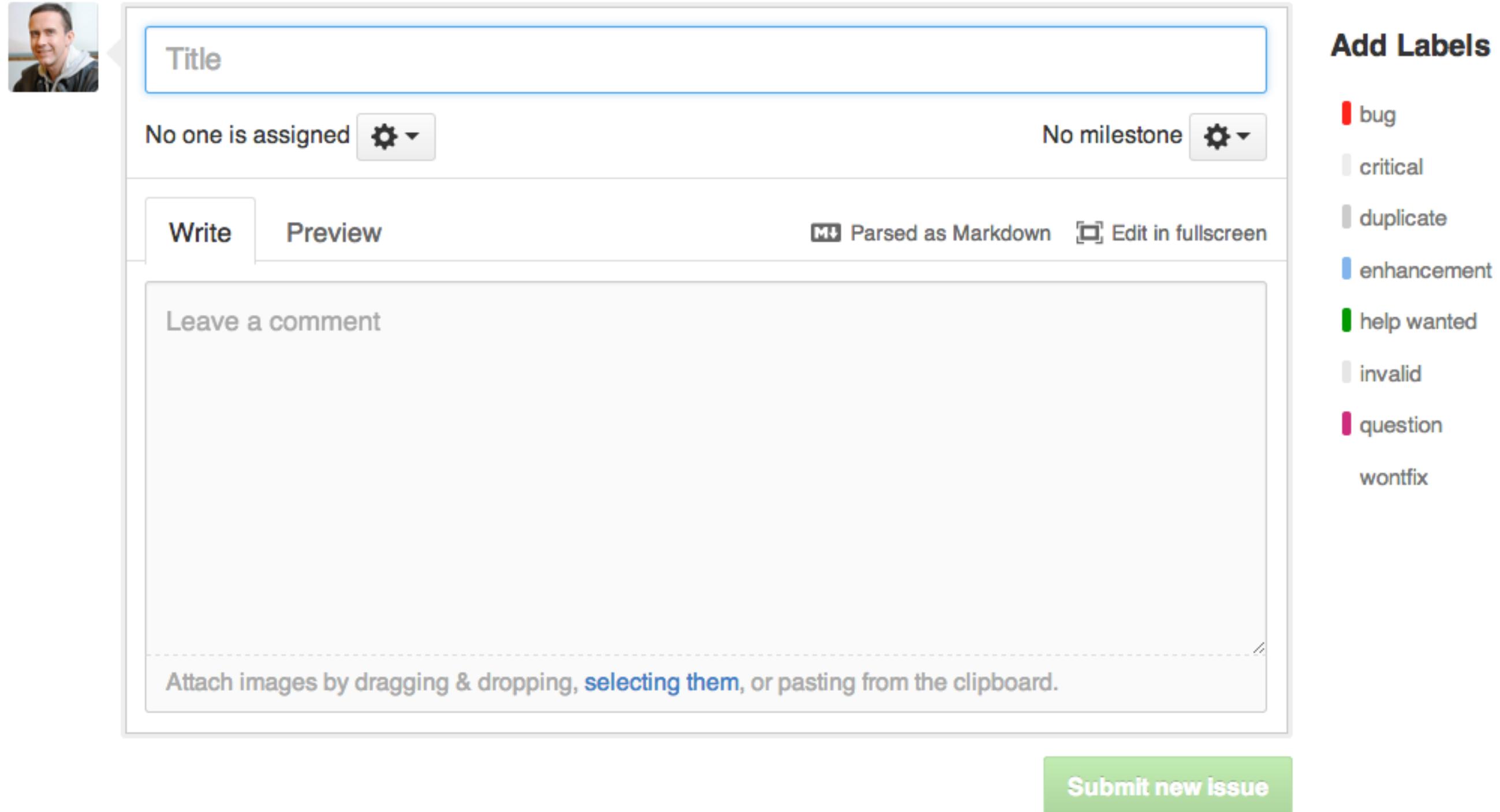
Creating an Issue

Click on “issues” in right hand nav



A screenshot of a GitHub issues page. At the top right, there is a green button labeled "New Issue". A hand-drawn style arrow points from the text "create new issue" to this button. The page shows filtering options: "Clear milestone and label filters", "1 Open", "0 Closed", and "Sort: Newest". Below these are buttons for "Close", "Label", "Assignee", and "Milestone". A single issue card is visible, labeled "#1", with the title "Fix IE3 critical" and the note "Opened by PeterBell Wednesday at 11:36am".

Creating an Issue



The screenshot shows the GitHub issue creation interface. At the top left is a user profile picture. Below it is a large input field labeled "Title". Underneath the title field are two status indicators: "No one is assigned" with a gear icon and "No milestone" with a gear icon. To the right of these is a "Add Labels" sidebar containing a list of labels with corresponding color swatches: bug (red), critical (light gray), duplicate (gray), enhancement (blue), help wanted (green), invalid (light gray), question (purple), and wontfix (pink). Below the status indicators are two tabs: "Write" (selected) and "Preview". To the right of the tabs are two buttons: "Parsed as Markdown" and "Edit in fullscreen". The main body of the form is a large text area labeled "Leave a comment". At the bottom of this area is a note: "Attach images by dragging & dropping, [selecting them](#), or pasting from the clipboard.". At the bottom right of the form is a green "Submit new Issue" button.

Issue Notifications

Difficult to use without a keyboard or mouse #3

The screenshot shows a GitHub issue page for a comment made by PeterBell. The page includes standard issue metadata like Owner, Labels, Milestone, and Assignee. A prominent 'Notifications' section is displayed, containing two large buttons: 'Subscribe' and 'Unsubscribe'. Handwritten annotations include a callout pointing to the 'Unsubscribe' button with the text 'You can unsubscribe from an issue.' Another callout points to the 'Subscribe' button with the text 'Or subscribe to an issue'.

Open PeterBell opened this issue 5 minutes ago · 0 comments

PeterBell commented
No description provided

PeterBell commented

Notifications

Unsubscribe

Subscribe

You don't receive notifications for this thread because you're ignoring it.

Owner

Labels None yet

Milestone No milestone

Assignee No one assigned

Notifications

Unsubscribe

You're receiving notifications because you authored the thread.

1 participant

Close Comment

Or subscribe to an issue

Responding to Issues by Email



We decided to "open the kimono" - we should share *all* of our rules. What do you think @deadlyvipers?

On Jun 5, 2014, at 3:32 PM, GitHub Sample Teacher <notifications@github.com> wrote:

I don't think we should give too many of our secrets away.

—
Reply to this email directly or [view it on GitHub](#).

You can reply to an email to respond to an issue.

Adding Commits to Issues

← Doesn't work on IE3 #1

! Open

PeterBell opened this issue 30 minutes ago · 0 comments



PeterBell commented 30 minutes ago

No description provided.

Owner



PeterBell referenced this issue from a commit 14 seconds ago



Should help with issue #1

f5a845d



Write

Preview



Parsed as Markdown



Edit in fullscreen

Leave a comment

UB

Mention an issue in a commit
and it'll show up in the issue.

```
$ git commit -m "Should help  
with issue #1"
```

Referencing Issues in Commits

Doesn't work on IE3 #1

 Closed

PeterBell opened this issue 31 minutes ago · 0 comments



PeterBell commented 31 minutes ago

No description provided.

Owner



 PeterBell referenced this issue from a commit a minute ago

 Should help with issue #1

f5a845d

 PeterBell closed this issue from a commit 14 seconds ago

 Fixes #1

Reference issues from commits
to tie work together.

7e378d1



PeterBell closed this in 7e378d1 a minute ago

HUB

Issues - keyboard shortcuts

3 Open 0 Closed Sort: Newest ▾

Close Label Assignee Milestone

- ⓘ Difficult to use without a keyboard or mouse
Opened by PeterBell 3 minutes ago
- ⓘ Can't view the images on my flip phone
Opened by PeterBell 3 minutes ago
- ⓘ Doesn't work on IE3
Opened by PeterBell 4 minutes ago

Keyboard shortcuts available 

NO mouse required!

Keyboard shortcuts (Show all)

Site wide shortcuts

- s Focus command bar, search all repositories
- / Focus command bar, search the current repository
- g n Go to Notifications
- ? Bring up this help dialog
- j Move selection down
- k Move selection up
- x Toggle selection
- o / enter Open selection

Repositories

- g c Go to Code
- g i Go to Issues
- g p Go to Pull Requests
- g w Go to Wiki

Issues

- ⌘ enter Submit comment
- ⌘ shift p Preview comment
- ⌘ shift l Go fullscreen
- c Create issue
- l Create label
- u Back to issues
- / Focus issues search
- r Reply (quoting selected text)

Issues

The screenshot shows the GitHub Issues page for the repository `deadlyvipers/dojo_rules`. The repository is public, as indicated by the "PUBLIC" badge. The page displays 0 open issues and 8 closed issues, sorted by newest. There are sections for "Everyone's Issues", "Assigned to you", "Created by you", and "Mentioning you". A sidebar on the right contains various navigation links and icons. The URL in the browser is https://github.com/deadlyvipers/dojo_rules/issues?page=1&state=open.

Issues - deadlyvipers/dojo_rules

GitHub, Inc. [US] https://github.com/deadlyvipers/dojo_rules/issues?page=1&state=open

This repository Search or type a command Explore Gist Blog Help PeterBell + × F

PUBLIC deadlyvipers / dojo_rules Unwatch 1 Star 0 Fork 2

Browse Issues Milestones New Issue

Everyone's Issues 0 0 Open 8 Closed Sort: Newest

Category	Count
Assigned to you	0
Created by you	0
Mentioning you	0

No issues to show. [Create a new issue.](#)

No milestone selected

Labels

Label	Count
bug	0
duplicate	0
enhancement	0
help wanted	0
invalid	0
question	0
wontfix	0

Issues

We're going to need more rails/rails GitHub, Inc. [US] https://github.com/rails/rails/ PeterBell

PUBLIC rails / rails Watch 1,510 Star 22,021 Fork 8,118

Ruby on Rails <http://rubyonrails.org>

44,371 commits 37 branches 210 releases 2,317 contributors

branch: master rails / +

call `capture` fewer times from `form_for` tenderlove authored 2 hours ago latest commit e2a97adbae

actionmailer Feature requests should be made on the mailing list, not submitted to a day ago

actionpack Feature requests should be made on the mailing list, not submitted to a day ago

actionview call `capture` fewer times from `form_for` 2 hours ago

activemodel Feature requests should be made on the mailing list, not submitted to a day ago

activerecord Merge pull request #15475 from sgrif/sg-initialize-attributes 4 hours ago

activesupport drastically reduce object allocations 2 hours ago

ci Reorganize ActiveRecord tasks [Arun Agrawal & Abd ar-Rahman Hamidi] 17 days ago

guides Fix typo in ActionController Overview 5 hours ago

railties File.exist? instead of File.exists? in bin/setup 6 hours ago

tasks Fix release task 2 months ago

Code Issues 606 Pull Requests 354

Pulse Graphs Network

HTTPS clone URL <https://github.com>

You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop Download ZIP

Wiki's

Level 4: Section 2

Why Use Wiki's?

- Start with a README.md
- When it becomes unwieldy, add a Wiki

Enabling Wiki's for a Project

Go to: settings/options

check box to implement wikis

Features

Wikis

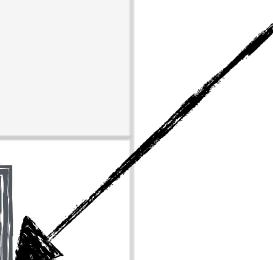
GitHub Wikis are the simplest way to let others contribute content. Any GitHub user can create and edit pages to use for documentation, examples, support, or anything you wish.

Restrict editing to collaborators only

Public wikis will still be readable by everyone.

Issues

GitHub Issues adds lightweight issue tracking tightly integrated with your repository. Add issues to milestones, label issues, and close & reference issues from commit messages.



limit editing to collaborators

Getting Started with Wiki's

PUBLIC



PeterBell / hellogitworld-1

Unwatch 1

Star 0

Fork 0



Welcome to the hellogitworld-1 wiki!

Wikis provide a place in your repository to lay out the roadmap of your project, show the current status, and document software better, together.

Create the first page



Creating a Wiki Page

Home

Write Preview

h1 h2 h3 Edit mode: Markdown

Welcome to the test_repo wiki!

GitHub flavored markdown

Linking Wiki Pages

The screenshot shows a GitHub wiki page titled "PeterBell / test_repo". The page content includes a "Welcome to the test_repo wiki" message and a link placeholder "[Contact Details](ContactDetails)". A modal dialog box titled "Insert Link" is open, prompting for "Link Text" (containing "ContactDetails") and "URL" (also containing "ContactDetails"). The "OK" button is visible at the bottom right of the dialog.

PUBLIC PeterBell / test_repo

Home

Write Preview

h1 h2 h3

Welcome to the test_repo wiki

[Contact Details](ContactDetails)

Insert Link

Link Text

ContactDetails

URL

ContactDetails

OK Cancel

Adding Markdown to a Wiki



Heading

Subheading

- * Bullet 1
- * Bullet 2
- * Bullet 3

buttons for formatting

just generate simple markdown

Adding Markdown to a Wiki

[Writing on GitHub](#) / GitHub Flavored Markdown

Article last updated on 07-Apr-14

How can we help?



GitHub Flavored Markdown

GitHub uses "GitHub Flavored Markdown," or GFM, across the site--in issues, comments, and pull requests. It differs from standard Markdown (SM) in a few significant ways, and adds some additional functionality.

If you're not already familiar with Markdown, take a look at [Markdown Basics](#). If you'd like to know more about features that are available in issues, comments, and pull request descriptions, such as task lists, read [Writing on GitHub](#).

<https://help.github.com/articles/github-flavored-markdown>

Wiki's - Adding Badges



```
[! [Dependency Status](https://gemnasium.com/rails/arel.svg)]
```

the source

<https://github.com/rails/arel/>

Wiki's

deadlyvipers/dojo_rules

GitHub, Inc. [US] https://github.com/deadlyvipers/dojo_rules

This repository Search or type a command Explore Gist Blog Help PeterBell + ⚔️ ↗

deadlyvipers / dojo_rules

Unwatch 1 Star 0 Fork 2

Description Website

Short description of this repository Website for this repository (optional) Save or cancel

5 commits 2 branches 2 releases 1 contributor

branch: master / +

Merge pull request #1 from PeterBell/master ...

PeterBell authored Monday at 2:30 PM latest commit df1d76604f

README.md Added README to the project 8 days ago

dojo_rules.md You should respect some Senseis 2 days ago

README.md

Code Issues Pull Requests Wiki Pulse Graphs Network Settings

GitHub Pages

Level 4: Section 3

Why Use GitHub Pages?

GitHub Pages can be used for:

- Adding documentation to a project
- Hosting a site for a user or organization

Creating Your User Page

- Create a repo called `username.github.io`
- Use the auto page generator
- Write content
- Pick theme
- Publish

Creating Your User Page

The screenshot shows a GitHub user profile page for the user 'deadlyvipers'. The page includes a profile picture, a repository named 'dojo_rules' (updated 26 minutes ago), a search bar, and buttons for filters, finding a repository, and creating a new repository. On the right, there are sections for 'Members' (listing 'PeterBell'), 'Teams' (with a search bar and 'Create a team' button), and 'Owners' (listing 1 member and 1 repository). The footer contains links for Terms, Privacy, Security, Contact, Status, API, Training, Shop, Blog, and About.

deadlyvipers

deadlyvipers

Filters Find a repository... + New repository

dojo_rules Updated 26 minutes ago ★ 0 2

Members 1 >

PeterBell Peter Bell

Add a member by username

Teams 1 >

Jump to a team

Owners 1 member · 1 repository

Create a team

© 2014 GitHub, Inc. Terms Privacy Security Contact Status API Training Shop Blog About

Creating a Project Page

- Select in settings
- Write content
- Select a layout
- Publish
- Content is in gh-pages branch

Creating a Project Page

The screenshot shows a GitHub repository page for the repository `deadlyvipers / dojo_rules`. The page includes a description input field, a website URL input field, and a save button. Below these are statistics: 5 commits, 1 branch, 2 releases, and 1 contributor. A dropdown menu shows the current branch is `master`. A merge pull request from `PeterBell/master` is listed. The repository contains files `README.md` and `dojo_rules.md`. The `README.md` file content is displayed as "Dojo Rules". On the right side, there is a sidebar with links for Issues (1), Pull Requests (0), Wiki, Pulse, Graphs, Network, and Settings. The Settings link is currently being interacted with, as indicated by a mouse cursor icon.

PUBLIC **deadlyvipers / dojo_rules**

Description Unwatch **1** Website Star **0** Fork **2**

Website

Short description of this repository

Website for this repository (optional)

Save or cancel

5 commits 1 branch 2 releases 1 contributor

branch: **master** / +

Merge pull request #1 from PeterBell/master ...

PeterBell authored Monday at 2:30 PM latest commit df1d76604f

README.md Added README to the project 9 days ago

dojo_rules.md You should respect some Senseis 3 days ago

README.md

Dojo Rules

Issues 1

Pull Requests 0

Wiki

Pulse

Graphs

Network

Settings

HTTPS clone URL

<https://github.com/>

Configuring a Custom Domain

- Create “CNAME” file in root of repo
- Should be on same branch as Pages content
- Configure CNAME with DNS host

Configuring a Custom Domain

The screenshot shows a GitHub repository page for the user 'deadlyvipers' named 'deadlyvipers.github.io'. The repository is public and has 1 commit, 1 branch, 0 releases, and 1 contributor. The 'branch: master' dropdown is set to 'deadlyvipers.github.io / +'.

Description: Short description of this repository

Website: Website for this repository (optional)

Save or cancel

Code

Issues: 0

Pull Requests: 0

Wiki

Pulse

Graphs

Network

Settings

HTTPS clone URL: <https://github.com/deadlyvipers/deadlyvipers.github.io>

We recommend [adding a README](#) to this repository to help give people an overview of your project.

[Add a README](#)

Clone in Desktop

GitHub, Inc. [US] https://github.com/deadlyvipers/deadlyvipers.github.io

PUBLIC deadlyvipers / deadlyvipers.github.io

Unwatch 1 Star 0 Fork 0

Explore Gist Blog Help PeterBell

1 commit 1 branch 0 releases 1 contributor

branch: master + deadlyvipers.github.io / +

Replace master branch with page content via GitHub

PeterBell authored 11 minutes ago latest commit 6f173b299b

File	Content	Time
images	Replace master branch with page content via GitHub	11 minutes ago
javascripts	Replace master branch with page content via GitHub	11 minutes ago
stylesheets	Replace master branch with page content via GitHub	11 minutes ago
index.html	Replace master branch with page content via GitHub	11 minutes ago
params.json	Replace master branch with page content via GitHub	11 minutes ago



• MASTERING • GITHUB



• MASTERING • GITHUB

Configuring a New Repository

Level 5: Section 1

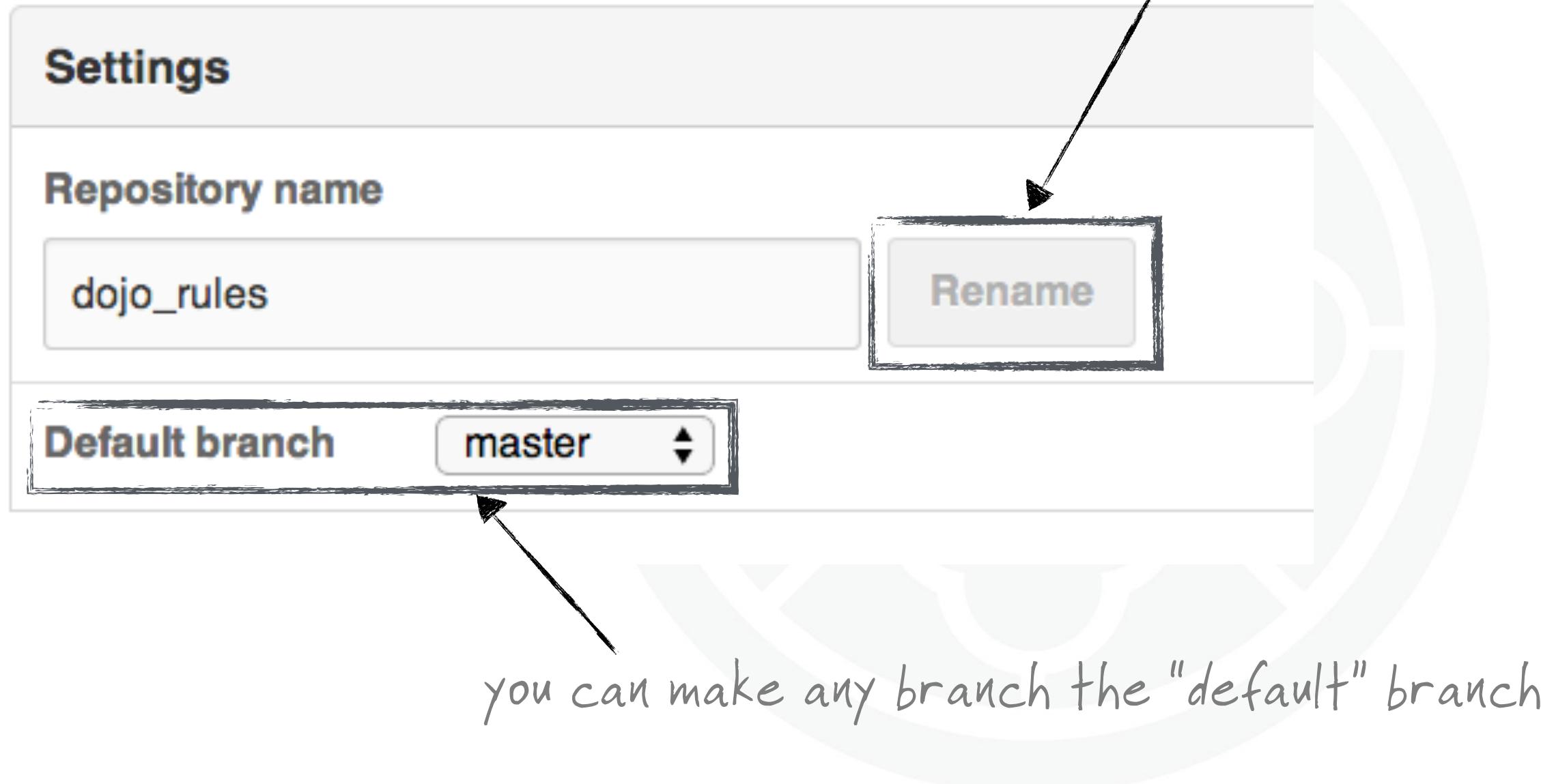
Options

Under settings:



High Level Options

Under settings - options



Configuring Features

Under settings - options

Features

Wikis
GitHub Wikis are the simplest way to let others contribute content. Any GitHub user can create and edit pages to use for documentation, examples, support, or anything you wish.

Restrict editing to collaborators only
Public wikis will still be readable by everyone.

Issues
GitHub Issues adds lightweight issue tracking tightly integrated with your repository. Add issues to milestones, label issues, and close & reference issues from commit messages.

it's easy to switch on and off support for Wiki's and issues

Configuring GitHub Pages

Under settings - options

GitHub Pages

Changes may take up to **ten minutes** to be visible.

Your site is published at http://peterbell.github.io/dojo_rules2

Update your site

Easily change your content or theme with the page generator.

To publish a page manually, push an HTML or **jekyll** site to your gh-pages branch. [More info.](#)

Automatic page generator

the automatic page generator is the easiest way to add GitHub pages to a repo

it creates a gh-pages branch with the content

The Danger Zone!

Under settings - options

Danger Zone

Make this repository private

Public forks can't be made private. Please [duplicate the repository](#) or [contact support](#).

[Make private](#)

Transfer ownership

Transfer this repo to another user or to an organization where you have admin rights.

[Transfer](#)

Delete this repository

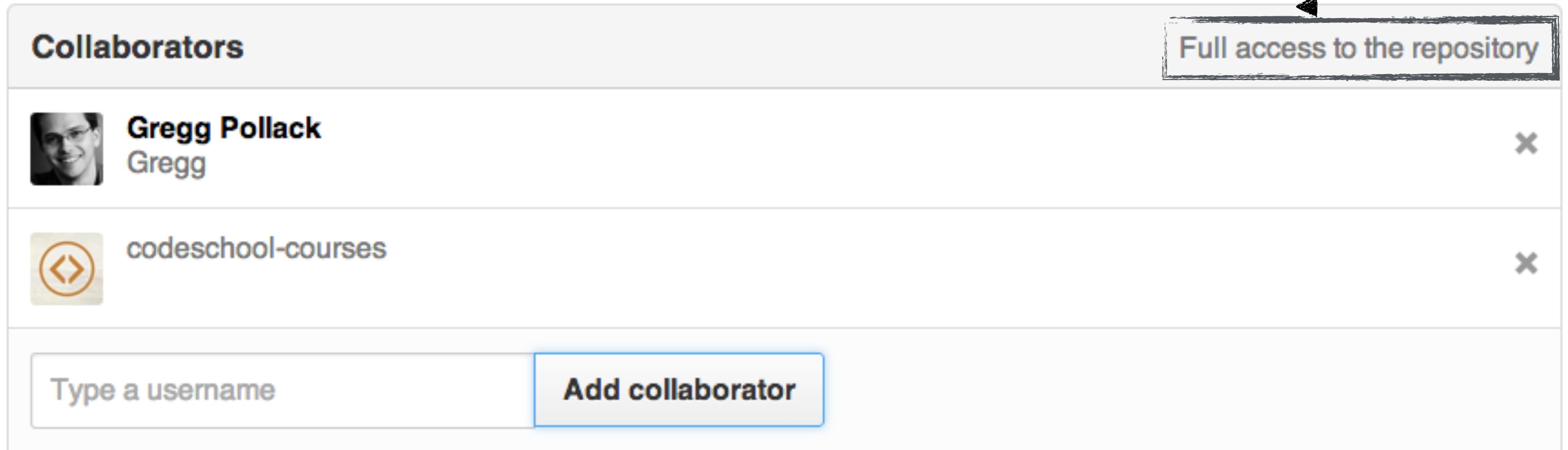
Once you delete a repository, there is no going back. Please be certain.

[Delete this repository](#)

Managing Collaborators

Under settings - collaborators

except for administering



The screenshot shows the 'Collaborators' section of a GitHub repository settings page. It lists two collaborators: Gregg Pollack (with a photo) and codeschool-courses (with a logo). Both have an 'X' icon to their right. Below the list is a search bar labeled 'Type a username' and a blue 'Add collaborator' button.

Collaborators

 **Gregg Pollack**
Gregg

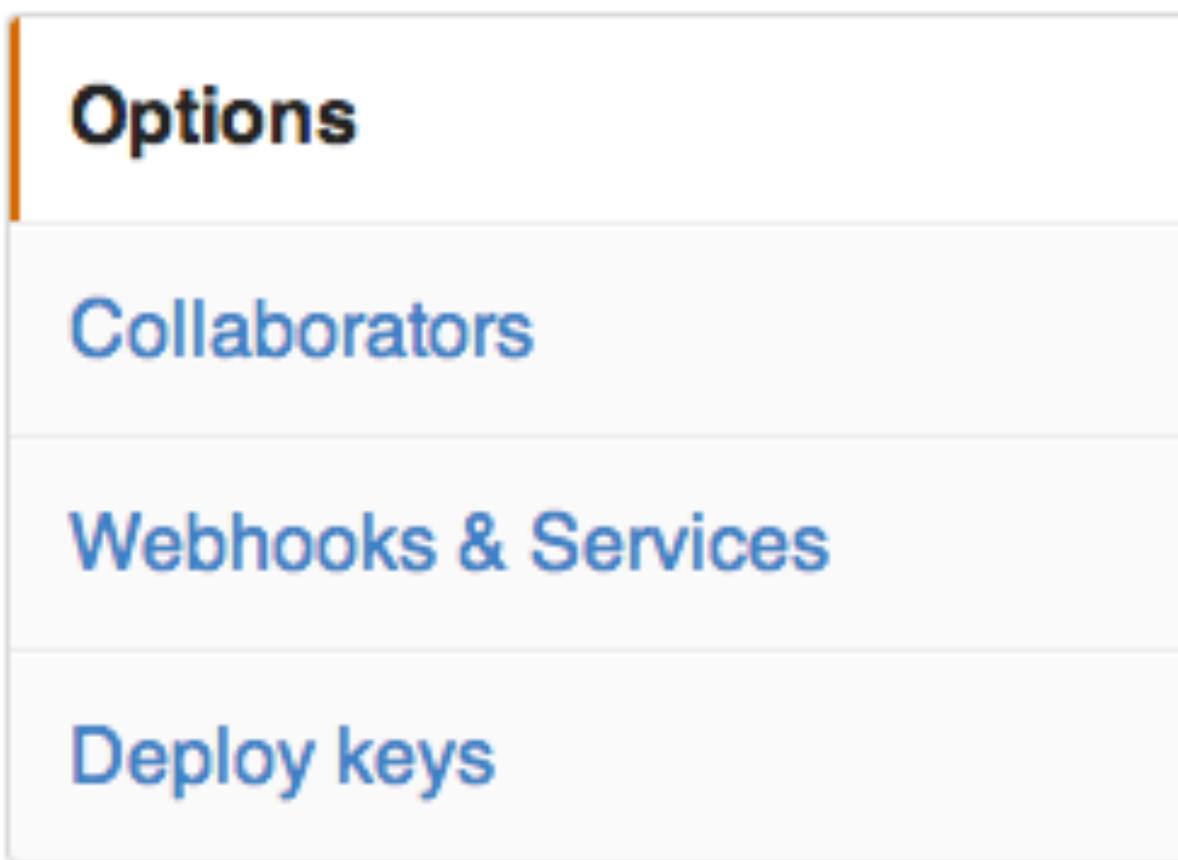
 codeschool-courses

Type a username **Add collaborator**

Full access to the repository

Other Options

Under settings:



Configuring a New Repository

The screenshot shows a GitHub user profile for 'deadlyvipers'. The top navigation bar includes links for Explore, Gist, Blog, and Help. A search bar and a command input field are also present. On the right, a user profile for 'PeterBell' is shown.

The main area displays two repositories: 'deadlyvipers.github.io' (CSS, 0 stars, 0 forks) and 'dojo_rules' (0 stars, 2 forks). A '+ New repository' button is located below the search bar.

On the right side, there are two sections: 'Members' (listing 'PeterBell') and 'Teams' (listing 'Owners' with 1 member and 2 repositories). A 'Create a team' button is also present in the Teams section.

At the bottom, standard GitHub footer links are provided: Status, API, Training, Shop, Blog, and About.

Page URL: https://github.com/deadlyvipers

Repository Details:

- deadlyvipers.github.io**: Updated 9 hours ago, CSS, 0 stars, 0 forks
- dojo_rules**: Updated 9 hours ago, 0 stars, 2 forks

Members:

- PeterBell (Peter Bell)

Teams:

- Owners**: 1 member · 2 repositories

Bottom Navigation:

- Status API Training Shop Blog About

Managing a Project via GitHub

Start with the README.md

The contents of the README.md display on the home page

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

 README.md

Welcome to Rails

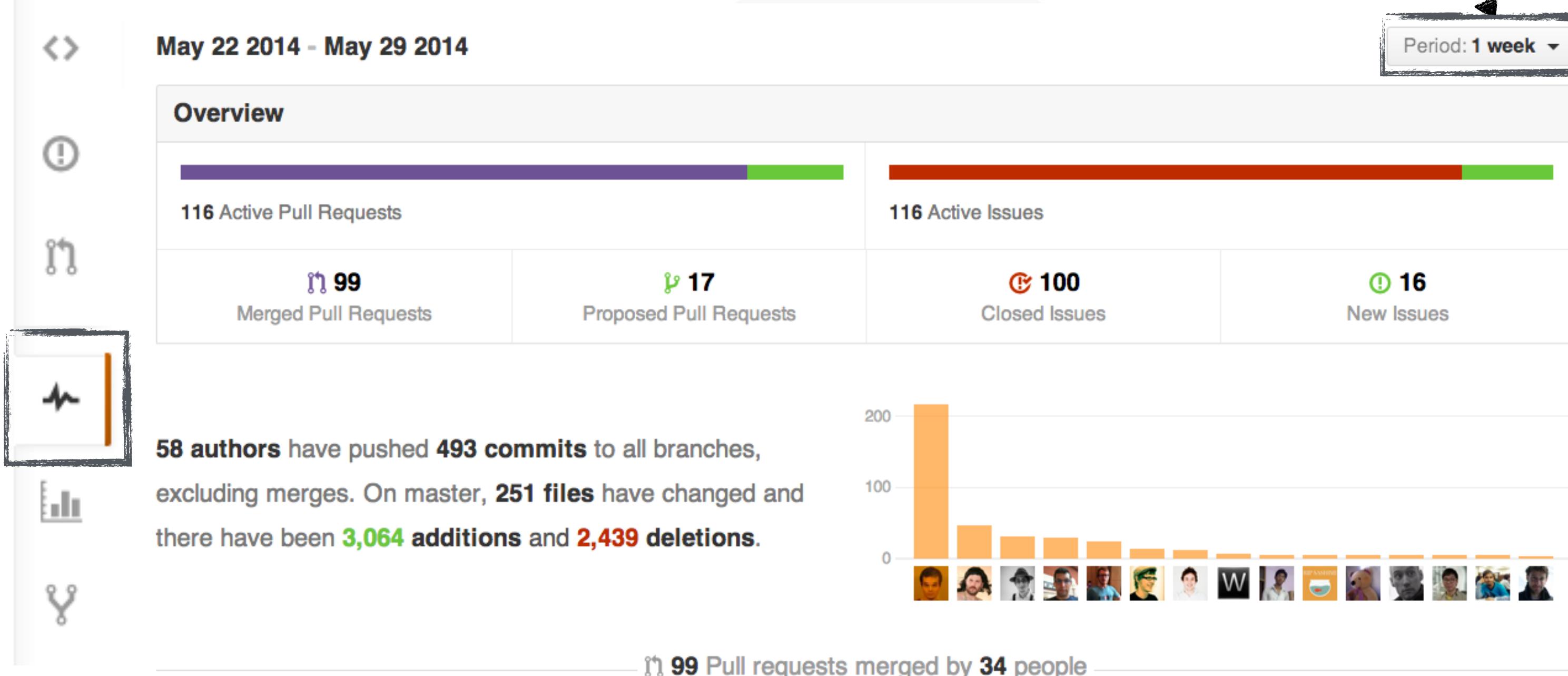
Rails is a web-application framework that includes everything needed to create database-backed web applications according to the [Model-View-Controller \(MVC\)](#) pattern.

Understanding the MVC pattern is key to understanding Rails. MVC divides your application into three layers, each with a specific responsibility.

Checking the Pulse

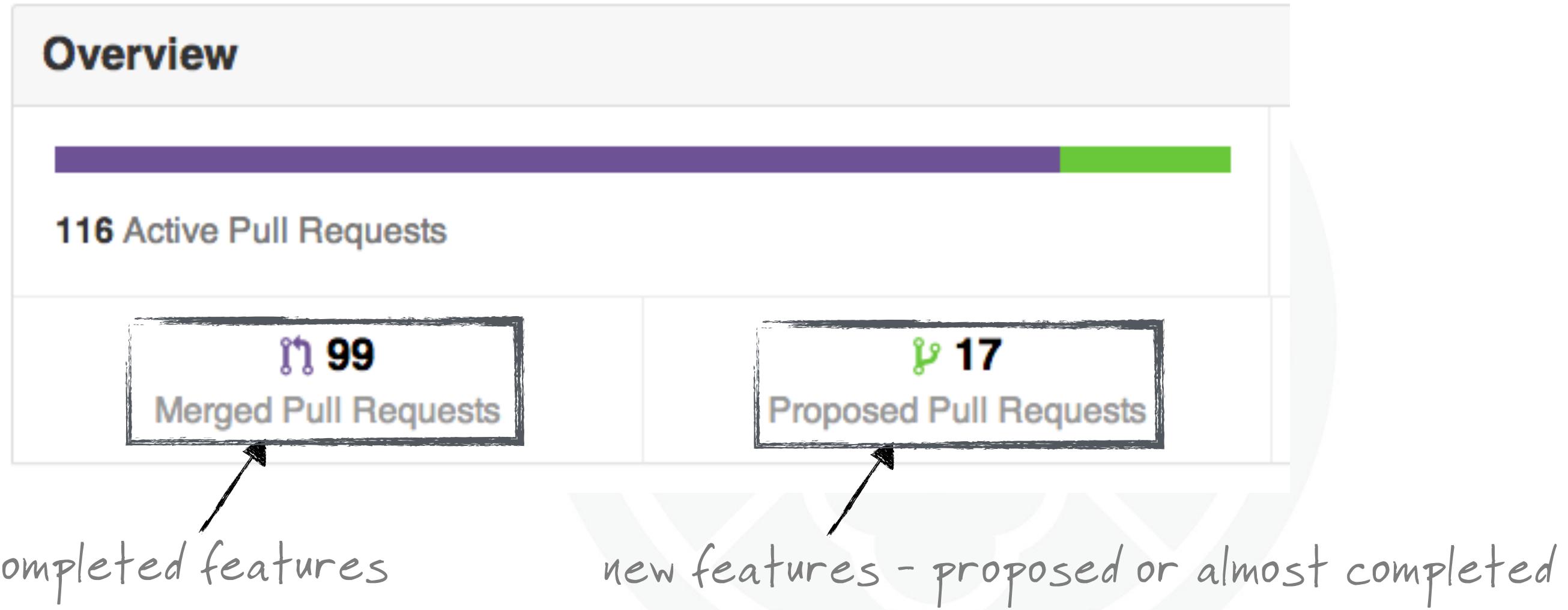
The pulse tab gives an overview of recent activity

start by selecting a timeframe



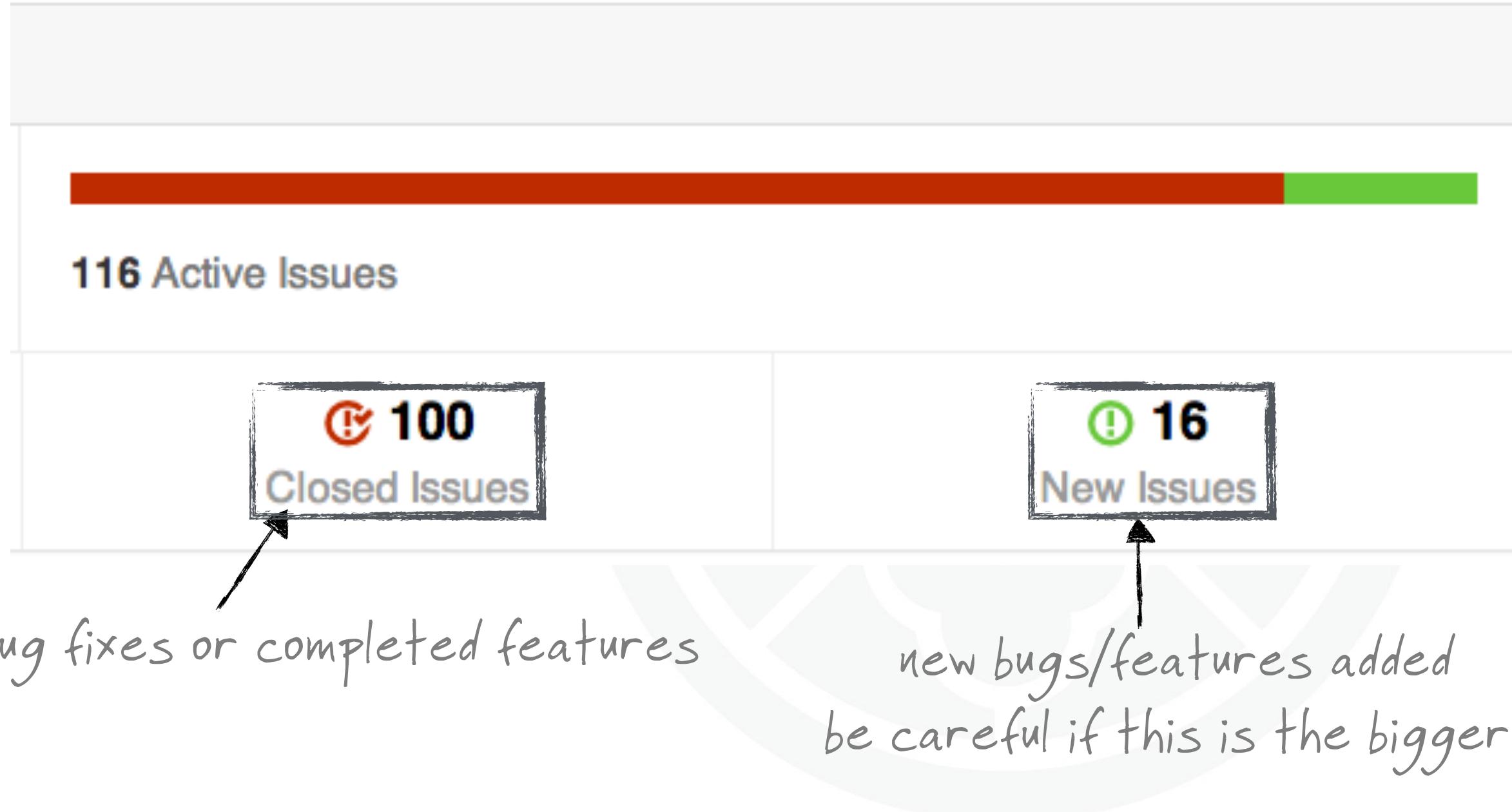
Checking the Pulse - Pull Request Stats

Pull requests give a sense of how many features were started/finished



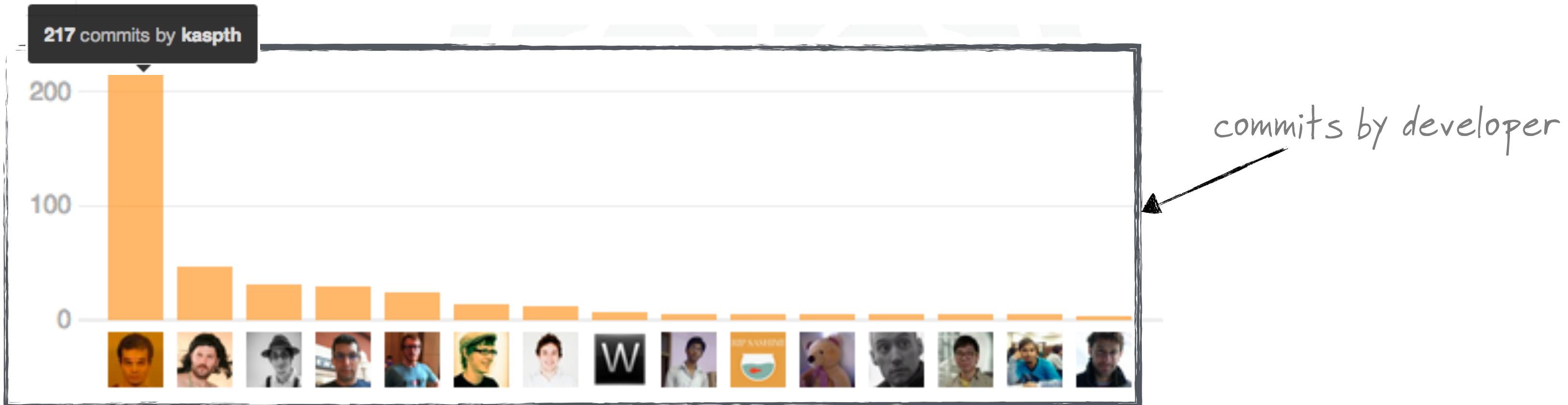
Checking the Pulse - Issue Stats

Issues give a sense of features or bugs being worked on



Checking the Pulse - By the Numbers

58 authors have pushed **493 commits** to all branches, excluding merges. On master, **251 files** have changed and there have been **3,064 additions** and **2,439 deletions**.



Checking the Pulse - Pull Request Listings

11 99 Pull requests merged by 34 people

Merged #15411 Fix broken link in Rails 3.0 release_notes 18 minutes ago

Merged #15398 Refactor serialized types to be partially defined as custom properties 41 minutes ago

Merged #15396 Correctly pass joins when creating a JoinDependency. 4 hours ago

Merged #15407 fixed ActiveRecord documentation [ci skip] 5 hours ago

11 17 Pull requests proposed by 12 people

Proposed #15262 Fix * ids method for has many with primary key Friday at 2:57am

Proposed #15266 If a counter_cache exists, use it for #empty? Friday at 6:47am

Proposed #15306 [WIP] Rails 4.2 release notes guide Saturday at 11:57pm

Checking the Pulse - Issue Listings

🕒 100 Issues closed by 22 people

Closed

#15408 ActiveRecord::Relation creating invalid SQL when using .select(x,y,z) and .size 4 hours ago

Closed

#15402 Flash messages isn't store in session 8 hours ago

Closed

#14730 Sprockets::Rails::Helper::AssetFilteredError in Static#home 9 hours ago

Closed

#15369 Duplicating hstore columns returns nil since Rails 4.1 9 hours ago

⌚ 16 Issues created by 16 people

Opened

#15409 Getting started With Engines eratta 2 hours ago

Opened

#15405 ActiveRecord fails, when querying multiple tables, passing a hash to order and using limit. 7 hours ago

Opened

#15394 ActiveRecord::Migration.maintain_test_schema! doesn't work with structure.sql yesterday at 5:04pm

Checking the Pulse - Conversation Listings

89 Unresolved conversations

Sometimes conversations happen on old items that aren't yet closed. Here is a list of all the Issues and Pull Requests with unresolved conversations.

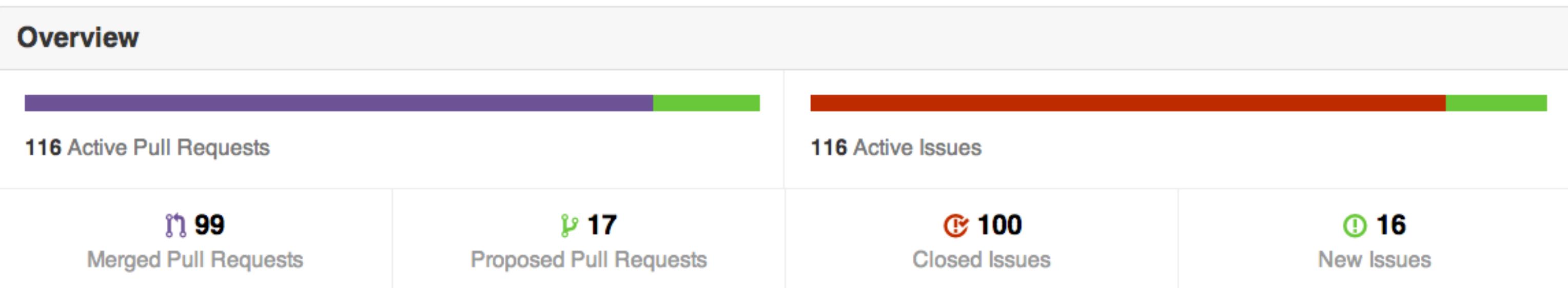
Open	#15098 Add PGRange class.	yesterday at 7:49am	41 new comments
Open	#15172 implement ActiveRecord::Base#pretty_print	5 hours ago	9 new comments
Open	#14117 Template streaming (render stream: true) does not work in latest Rails	Tuesday at 2:48pm	8 new comments
Open	#12036 Add increment_counter!, decrement_counter!, update_counters!	Tuesday at 7:15pm	5 new comments
Open	#14021 class_attribute not thread-safe	14 hours ago	4 new comments
Open	#13873 rails on live stream seems to expires with error on server	23 hours ago	4 new comments

Checking the Pulse - All About the Deltas

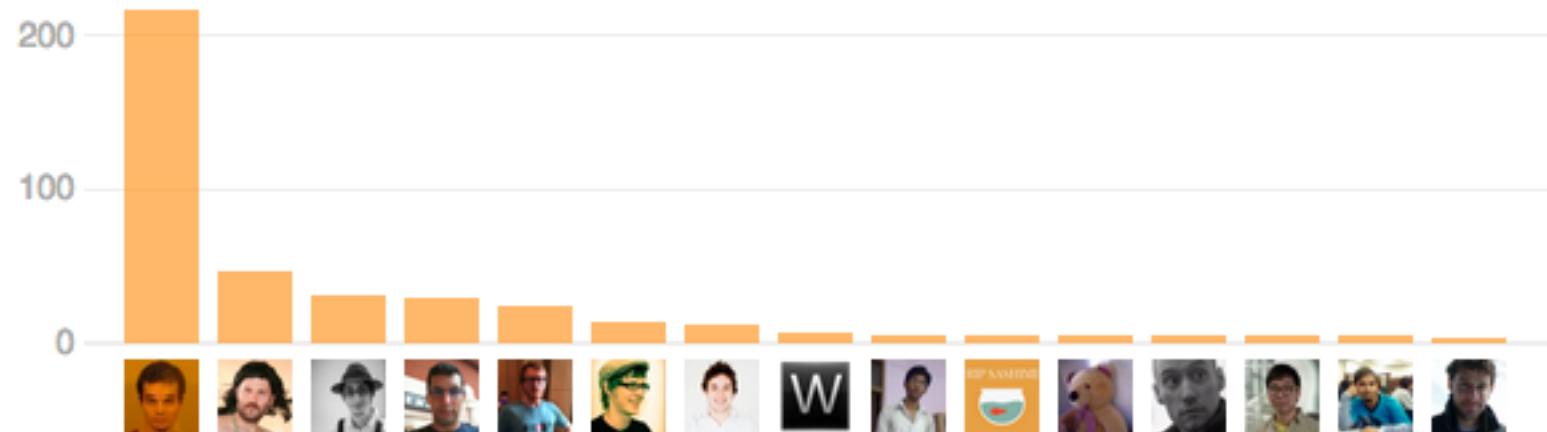
Pulse shows changes in numbers - indicating directionality

May 22 2014 - May 29 2014

Period: 1 week ▾



58 authors have pushed 493 commits to all branches, excluding merges. On master, 251 files have changed and there have been 3,064 additions and 2,439 deletions.



Graphs and Progress

Graphs show longer term trends



Contributors

Contributions over time



Commit activity

Commit activity

Commit activity over the previous year



Code frequency

Additions and deletions over time



Punch card

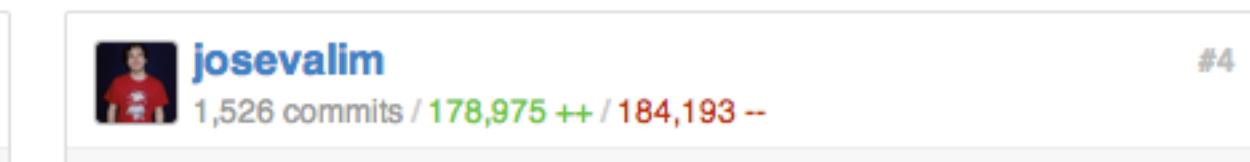
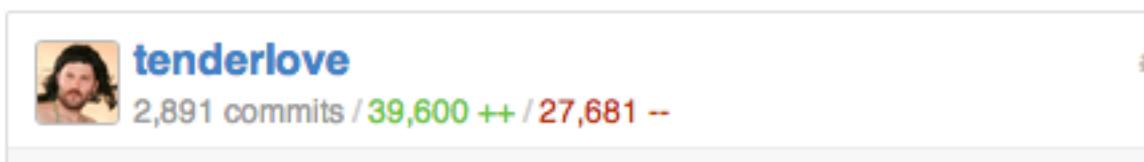
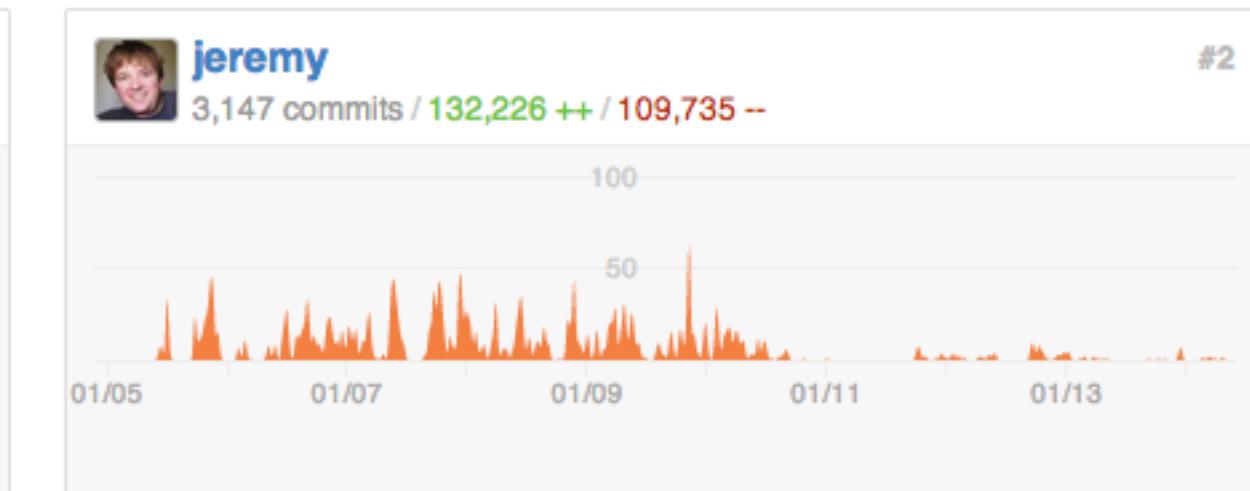
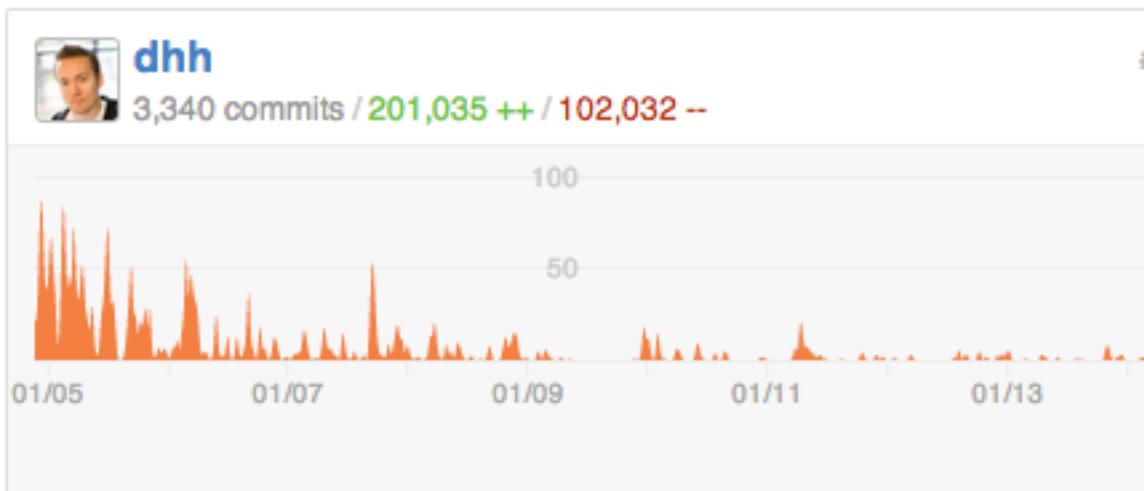
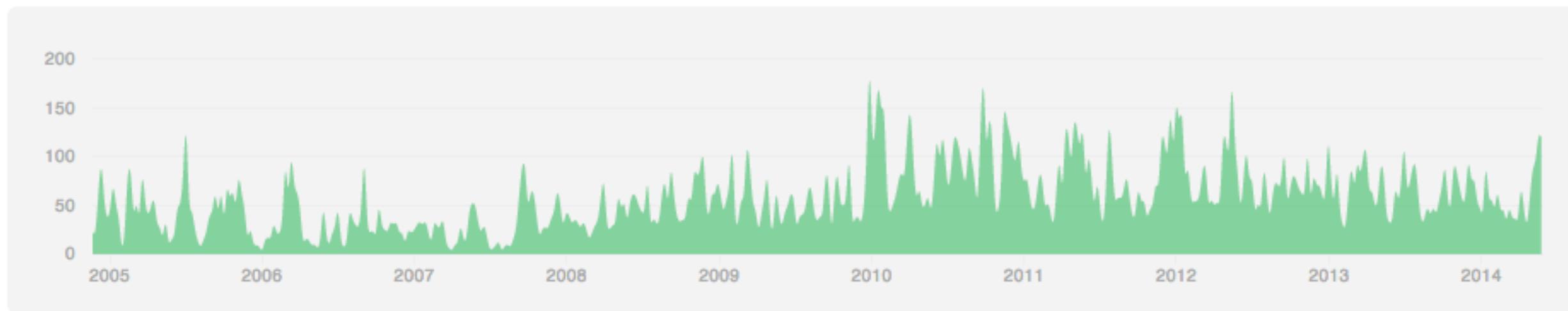
Time and day of commit activity

Graphs - Contributors to find MVPs

November 21st 2004 - May 30th 2014

Contributions to master, excluding merge commits

Contribution type: **Commits** ▾



Graphs - Contributors to find MVPs

November 21st 2004 - May 30th 2014

Contributions to master, excluding merge commits

Contribution type: **Commits** ▾

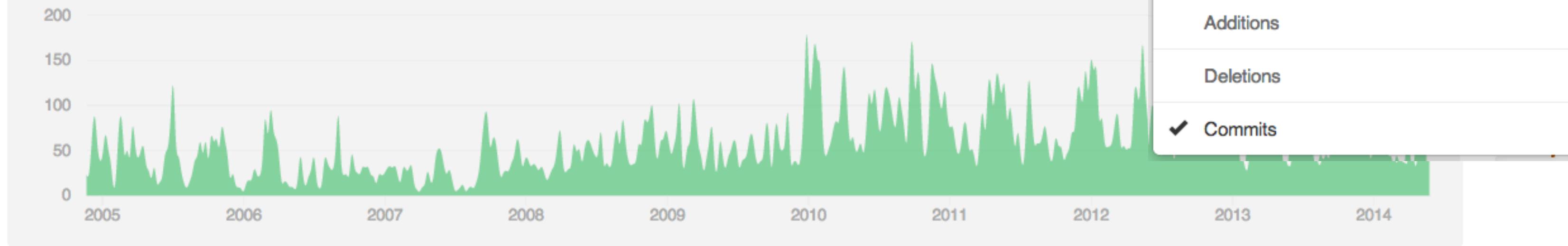


Filter & sort contributions

Additions

Deletions

Commits

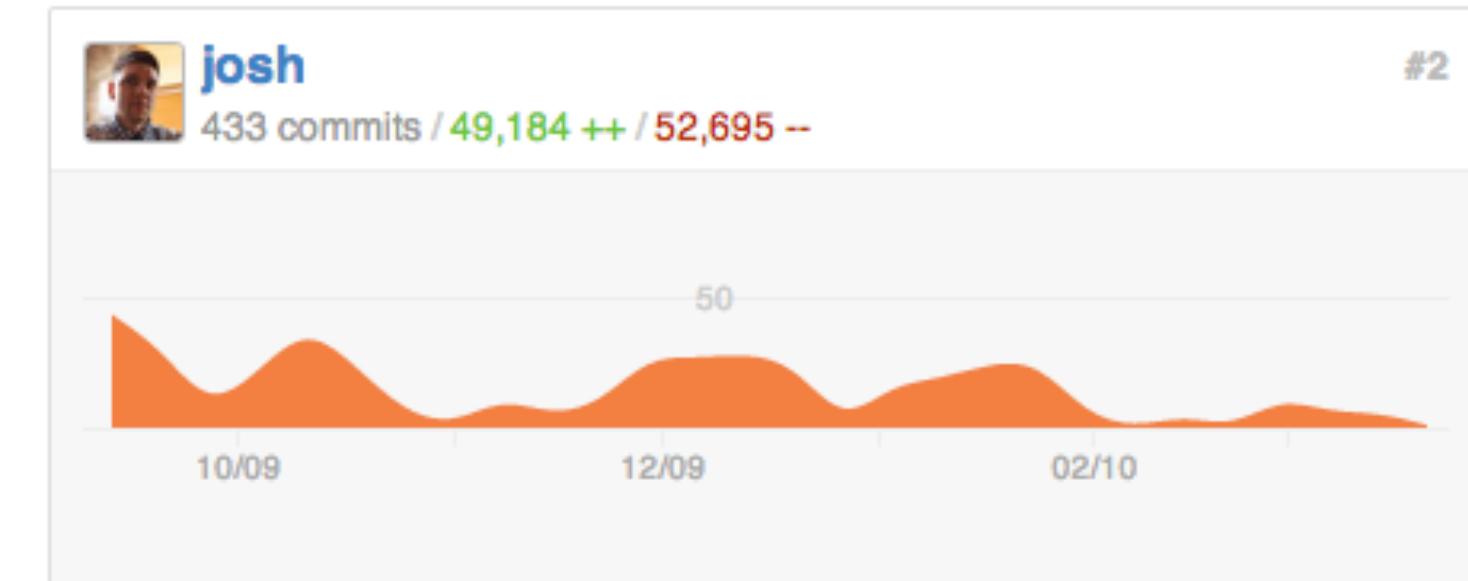
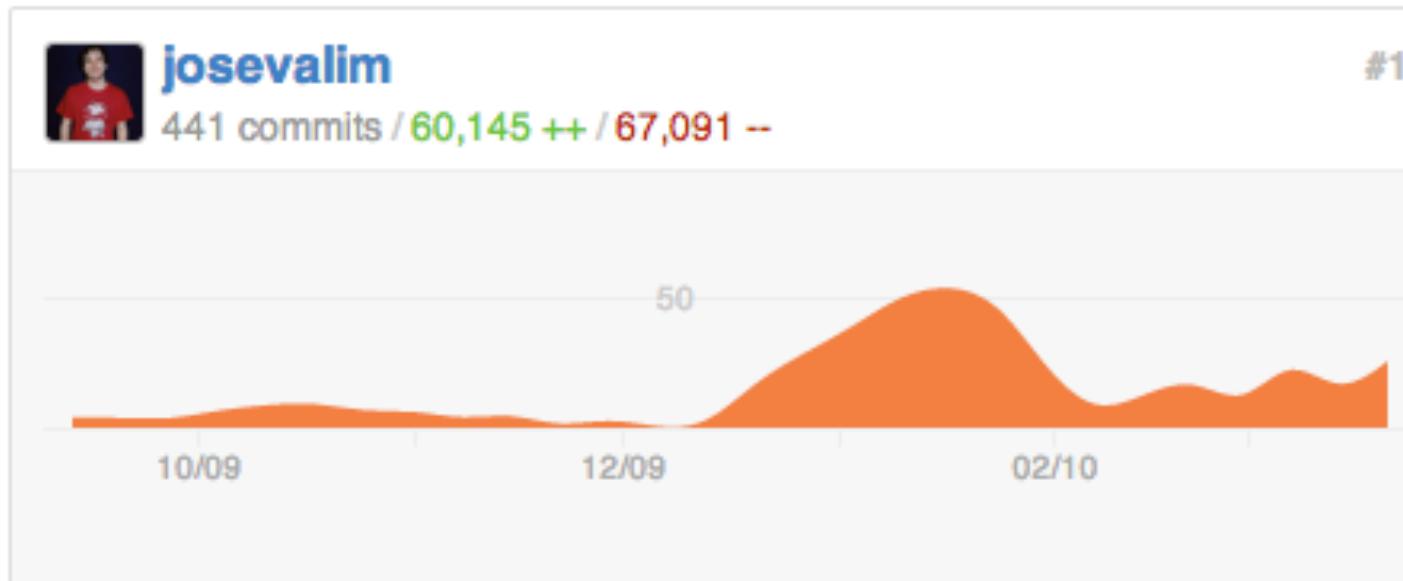
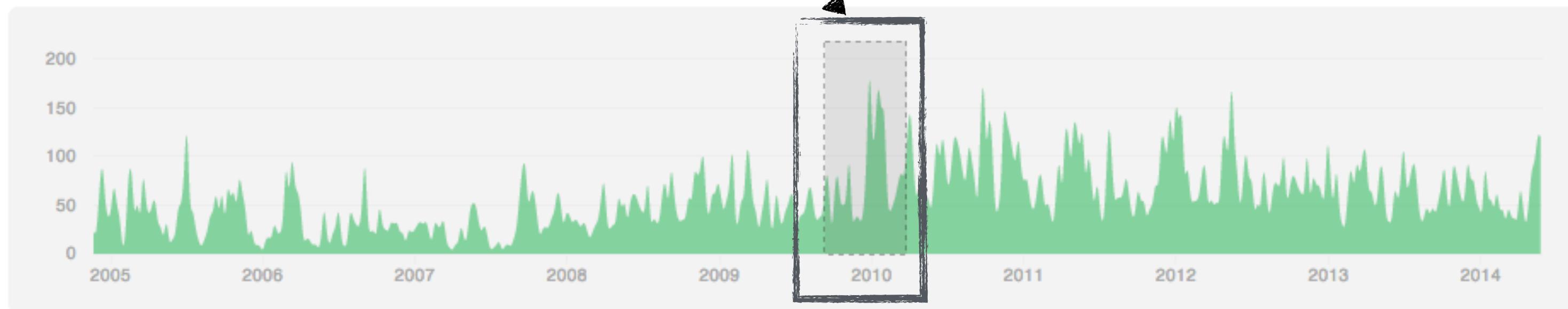


Graphs - Contributors identify MVPs

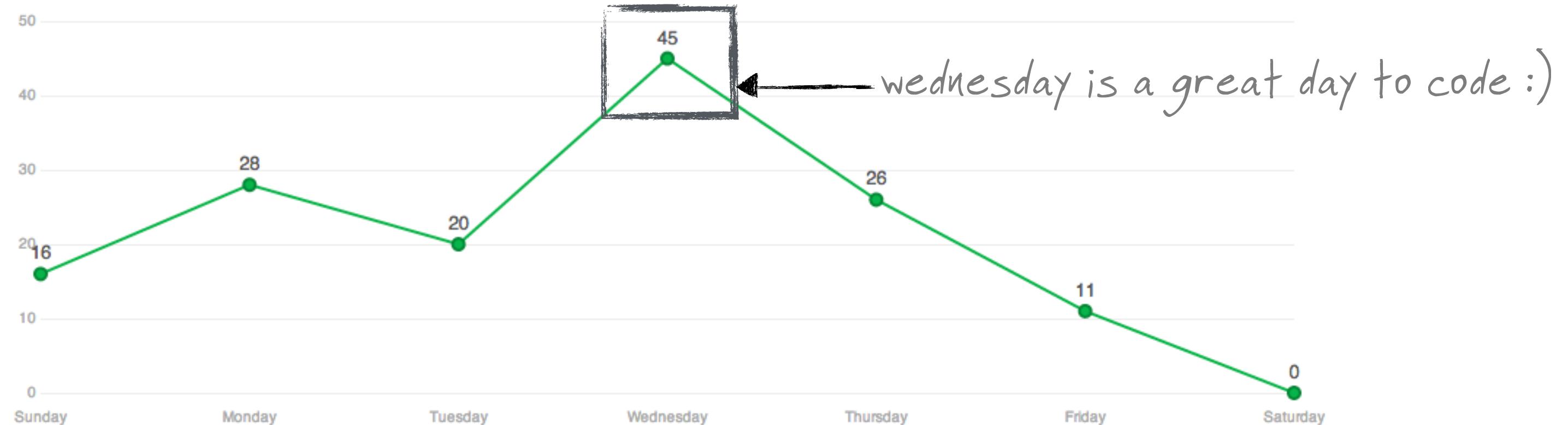
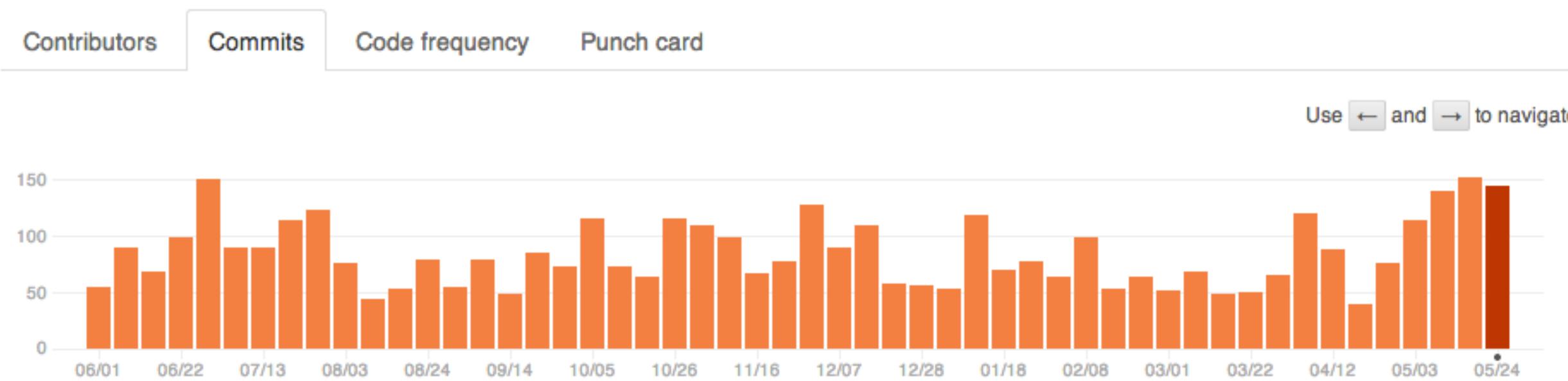
September 8th 2009 - March 24th 2010
Contributions to master, excluding merge commits

select a time period right on the graph

Contribution type: **Commits** ▾



Graphs - Commits shows busiest days/weeks



Graphs - Code Frequency Identifies Big Changes

Contributors

Commits

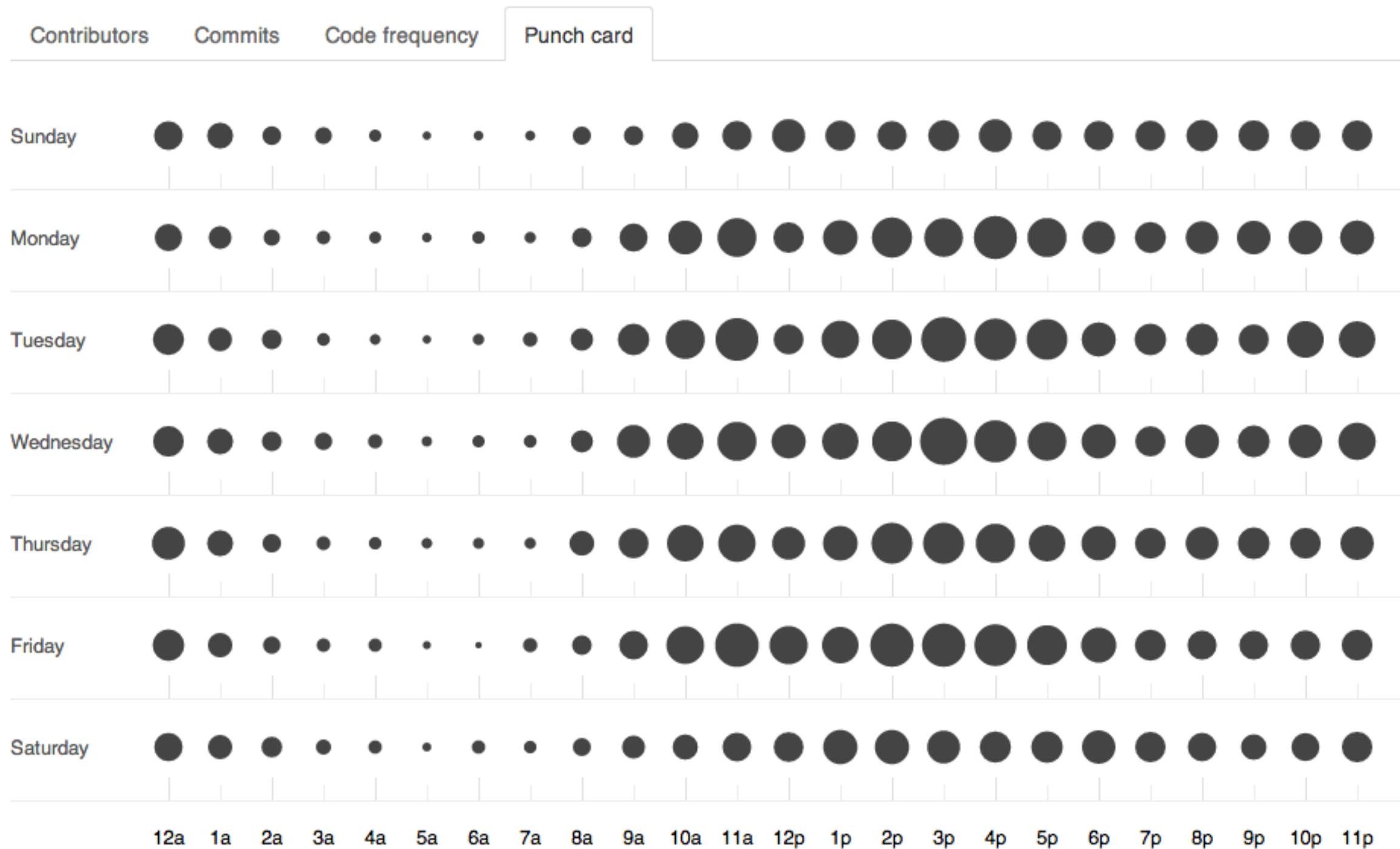
Code frequency

Punch card

Additions and Deletions per week



Graphs - Punchcards Showcase Productivity



The Absolute State of Issues

Browse Issues Milestones [New Issue](#)

Everyone's Issues 588

Created by you 0

Mentioning you 0

No milestone selected

Labels

Important	1
JRuby	5
PostgreSQL	24
actionmailer	4
actionpack	84
actionview	37
activemodel	12
activerecord	243
activesupport	56
...	-

588 Open 14,824 Closed

4.1 save callback touching DB fails due uninitialized relation delegate cache [#15422](#)
activerecord
Opened by kares 5 hours ago 2 comments

Fixed `ActiveSupport::TimeWithZone#` so precision is not unnecessarily lost [#15421](#)
Opened by gchan 6 hours ago 1 comment

simple validation to be able to run migrations with :table_name_with_underscore [#15419](#)
#15051
Opened by joseluistorres 12 hours ago

ActiveRecord created_at not set in before_create hook [#15418](#)
activerecord
Opened by everton 12 hours ago 8 comments

Consolidate duplicate to_formatted_s method [#15416](#)
Opened by jsgarvin 14 hours ago

Don't hang if streaming response is cancelled before completion [#15410](#)
actionpack
Opened by njakobsen 19 hours ago 5 comments

ActiveRecord fails, when querying multiple tables, passing a hash to order and using limit. [#15405](#)
activerecord
Opened by ushis yesterday at 10:06 AM 3 comments



The Absolute State of Pull Requests

All requests 342

Yours

Author	Count
bogdan	8
ccutrer	6
iantropov	5
estsauver	4
Empact	4
pftg	3
sgrif	3
kennyj	3
printercu	3
acapilleri	3
yangchenyun	3
JonRowe	3
Intrepidd	3

Opened Sort: Newest 1 2 3 ... 14 New pull request

ActiveSupport::TimeWithZone#` so precision is not unnecessarily lost #15421

Working with objects with a nanosecond component, the - method may unnecessarily cause loss ...
by gchan 6 hours ago gchan:time_with_zone_precision 1 comment

simple validation to be able to run migrations with :table_name_with_underscore #15419

This validation fix the issue with #15051 schema_migrations. I followed @raeno instructions to re...
by joseluistortes 12 hours ago joseluistortes:issue-with-primary-key-15051

Consolidate duplicate to_formatted_s method #15416

I found some duplicate code in the Date and Time core extensions and thought it would be easy eno...
by jsgarvin 14 hours ago jsgarvin:combine-common-date-and-time-to-fo

Don't hang if streaming response is cancelled before completion #15410

Because threads writing to the streaming response buffer aren't able to check if the buffer has b...
by njakobsen 19 hours ago njakobsen:patch-1 5 comments

Fix AS::NumberHelper results with rationals #15379

This change introduced a bug where Rational numbers are rounded always as if the :significant opt...
by xuanxu Wednesday at 6:50 AM xuanxu:rational_precision 1 comment

The Absolute State of Branches

branch: **4-0-stable** ▾ rails / Commits

May 30, 2014

 **Change the version schema to reflect how deprecations are handled** ... 268304f5cb +
rafaelfranca authored 22 minutes ago Browse code ➔

 **Merge pull request #11896 from nkondratyev/fix_pg_columns_for_distinct** ... c17c04f638 +
senny authored 5 hours ago Browse code ➔

4-0-6
Last updated Wednesday at 2:03 PM by zzak.

930 ahead
7,055 behind

 Compare

4-1-2
Last updated Wednesday at 2:03 PM by zzak.

364 ahead
1,973 behind

 Compare

3-2-stable
>Last updated on May 18 by tenderlove.

1,721 ahead
17,559 behind

 Compare

rf-experiment-private-api-documentation
Last updated on May 16 by chancancode.

3 ahead
589 behind

 Compare

3-0-test-i18n
Last updated on May 8 by carlosantoniodasilva.

1,626 ahead
26,633 behind

 Compare

GitHub Authentication/Authorization

Level 5: Section 2

Git Doesn't do Authentication

- User name and email are arbitrary strings
- Signed tags are the only auth option in Git

GitHub Doesn't Do Much Either

Impress your friends by getting DHH to commit to your project!



**David Heinemeier
Hansson**

dhh

david@basecamp.com

Joined on Mar 10, 2008

branch: **master** ▾

[dojo_rules2](#) / Commits

May 30, 2014



A commit from DHH

dhh authored 23 seconds ago

bda6fa45c9

[Browse code ➔](#)

```
$ git config --global user.email david@basecamp.com
$ touch dhh.txt
$ git add .
$ git commit -m "A commit from DHH"
$ git push
```

Works for anyone if you know their email!

Access Control in GitHub

- Owners can do anything
- Collaborators can do anything except manage settings
- Limit ability to commit using forks

GitHub Organizations and Teams

- Organizations are for code owned by groups
 - companies
 - OSS projects
 - etc.
- Teams allow easier management of permissions

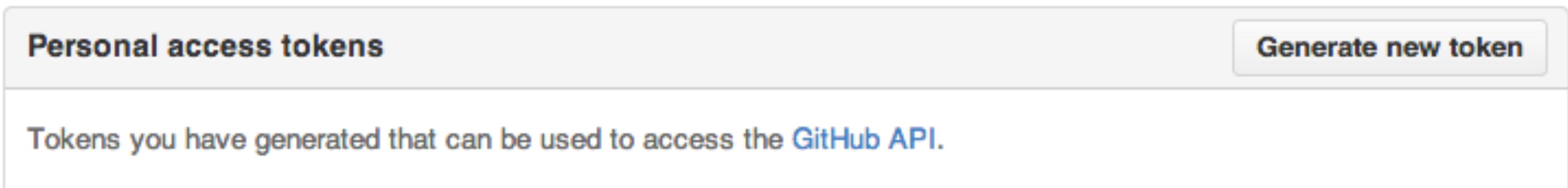
Connecting to GitHub

- HTTPS favored over SSH
- Use a credential helper to avoid typing passwords
- Upgrade to two factor authentication (2FA)

Creating Access Tokens

- For login with 2FA
- For access via build script/API calls, etc.

Edit Profile - Applications - Generate New Token



Creating Access Tokens

Applications / New personal access token

Token description

What's this token for?

Select scopes

Scopes *limit* access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo ⓘ	<input type="checkbox"/> repo:status ⓘ	<input type="checkbox"/> repo_deployment ⓘ
<input checked="" type="checkbox"/> public_repo ⓘ	<input type="checkbox"/> delete_repo ⓘ	<input checked="" type="checkbox"/> user ⓘ
<input type="checkbox"/> user:email ⓘ	<input type="checkbox"/> user:follow ⓘ	<input type="checkbox"/> admin:org ⓘ
<input type="checkbox"/> write:org ⓘ	<input type="checkbox"/> read:org ⓘ	<input type="checkbox"/> admin:public_key ⓘ
<input type="checkbox"/> write:public_key ⓘ	<input type="checkbox"/> read:public_key ⓘ	<input type="checkbox"/> admin:repo_hook ⓘ
<input type="checkbox"/> write:repo_hook ⓘ	<input type="checkbox"/> read:repo_hook ⓘ	<input checked="" type="checkbox"/> gist ⓘ
<input type="checkbox"/> notifications ⓘ		

Generate token

Automated GitHub Access Options

- SSH agent forwarding - great starting point, no automation available
- OAuth access tokens - same permissions as the user
- Deploy key - access token limited to a single repo
- Machine user - create a new GitHub account and manage its permissions

GitHub - Reviewing Security

Edit profile - security

Sessions

This is a list of devices that have logged into your account. Revoke any sessions that you do not recognize.

 ●	New York 199.47.73.122 Your current session	...
Chrome on OS X 10.9.3		
Location: New York, New York, United States		
Signed in: May 14, 2014		
 ●	Brooklyn 72.69.178.72 Last accessed on May 25, 2014	...
		Revoke

GitHub - Reviewing Security

Edit profile - security

Security history	
This is a security log of important events involving your account.	
repo.add_member – Add codeschool-courses to PeterBell/dojo_rules2	20 hours ago
repo.add_member – Add Gregg to PeterBell/dojo_rules2	20 hours ago
repo.rename – dojo_rules renamed to PeterBell/dojo_rules2	20 hours ago
repo.rename – test_repo renamed to PeterBell/test_repo2	20 hours ago
repo.create	22 hours ago
repo.destroy	22 hours ago
repo.create	22 hours ago
repo.destroy	22 hours ago
repo.create	22 hours ago
repo.destroy	22 hours ago
repo.create	22 hours ago

• MASTERING • GITHUB



• MASTERING • GITHUB

Integrating Existing Services

Level 6: Section 1

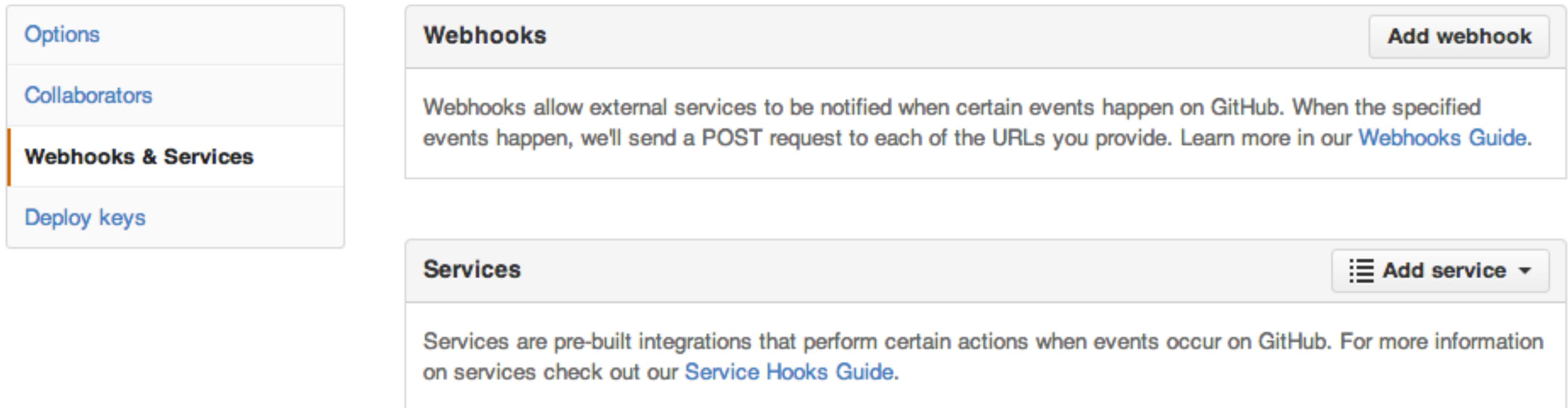
Integrating Existing Services

<https://github.com/github/github-services/tree/master/docs>

Adds Snap-CI to Github services.		
 rafbgarcia	authored on May 20	latest commit a2ddfc640e 
..		
 activecollab	update docs to new format. remove developer notes	2 years ago
 acunote	doc formatting	2 years ago
 agilebench	remove duplicated titles from config documentation	a month ago
 agilezen	remove duplicated titles from config documentation	a month ago
 amazonsns	remove duplicated titles from config documentation	a month ago
		
	config documentation	a year ago
	remove developer notes	a month ago
	Fixing some events.	2 years ago
		11 months ago
		9 months ago

Integrating Existing Services

Settings - Webhooks & Services



The screenshot shows the GitHub Settings interface for managing webhooks and services. On the left, a sidebar menu lists 'Options', 'Collaborators', 'Webhooks & Services' (which is selected and highlighted in blue), and 'Deploy keys'. The main content area is divided into two sections: 'Webhooks' and 'Services'. The 'Webhooks' section contains a descriptive text about what webhooks are and how they work, followed by a 'Add webhook' button. The 'Services' section contains a similar descriptive text about services and their purpose, followed by an 'Add service' button.

Options

Collaborators

Webhooks & Services

Deploy keys

Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen on GitHub. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

Services

Add service ▾

Services are pre-built integrations that perform certain actions when events occur on GitHub. For more information on services check out our [Service Hooks Guide](#).

Integrating Existing Services

The screenshot shows a GitHub repository page for the repository `deadlyvipers/dojo_rules`. The page includes the following elements:

- Header:** Shows the repository name `deadlyvipers/dojo_rules`, the URL `https://github.com/deadlyvipers/dojo_rules`, and navigation links for Explore, Gist, Blog, and Help.
- User Information:** Profile picture and name `PeterBell`.
- Repository Statistics:** PUBLIC, `deadlyvipers / dojo_rules`, Unwatch (1), Star (0), Fork (2).
- Description and Website:** Fields for entering a short description and website URL, with a `Save` button.
- Activity Metrics:** 5 commits, 2 branches, 2 releases, 1 contributor.
- Branch Selection:** A dropdown showing `branch: master` and a link to `dojo_rules / +`.
- Merge Pull Request:** A merge pull request from `PeterBell/master` to the current branch.
- Commit History:** A list of commits:
 - `PeterBell authored Monday at 2:30 PM`: Added `README.md` (8 days ago)
 - `You should respect some Senseis`: Added `dojo_rules.md` (2 days ago)
- File Content Preview:** Preview of the `README.md` file content.
- Section Header:**

Dojo Rules
- Page Footer:** A note stating the repository contains a list of dojo rules for the Deadly Vipers dojo.
- Right Sidebar:** Links to Issues (0), Pull Requests (0), Wiki, Pulse, Graphs, Network, and Settings, along with an HTTPS clone URL field.

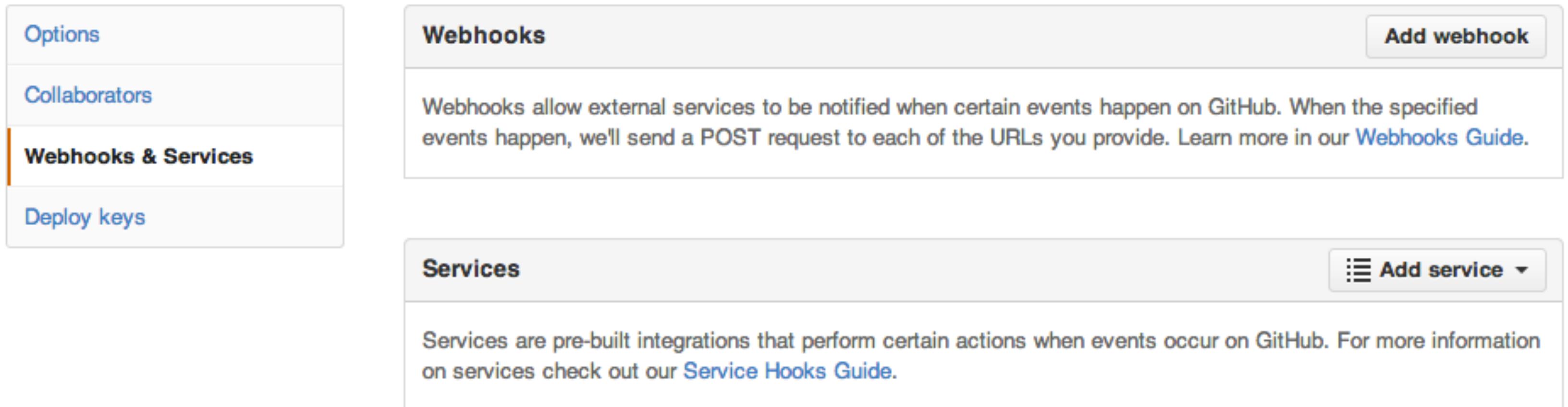
Integrations Vary in Sophistication

- **Basecamp** - list commits in progress page
- **JIRA** - resolve issues in commit messages
- **CircleCI** - runs tests automatically

Creating Custom Webhooks

Implementing Custom Webhooks

Settings - Webhooks & Services



The screenshot shows the GitHub settings interface for managing webhooks and services. On the left, a sidebar menu lists 'Options', 'Collaborators', 'Webhooks & Services' (which is selected and highlighted in blue), and 'Deploy keys'. The main content area is divided into two sections: 'Webhooks' and 'Services'. The 'Webhooks' section contains a descriptive paragraph about what webhooks are and how they work, followed by a 'Add webhook' button. The 'Services' section contains a similar descriptive paragraph about services and their purpose, followed by an 'Add service' button.

Webhooks

Add webhook

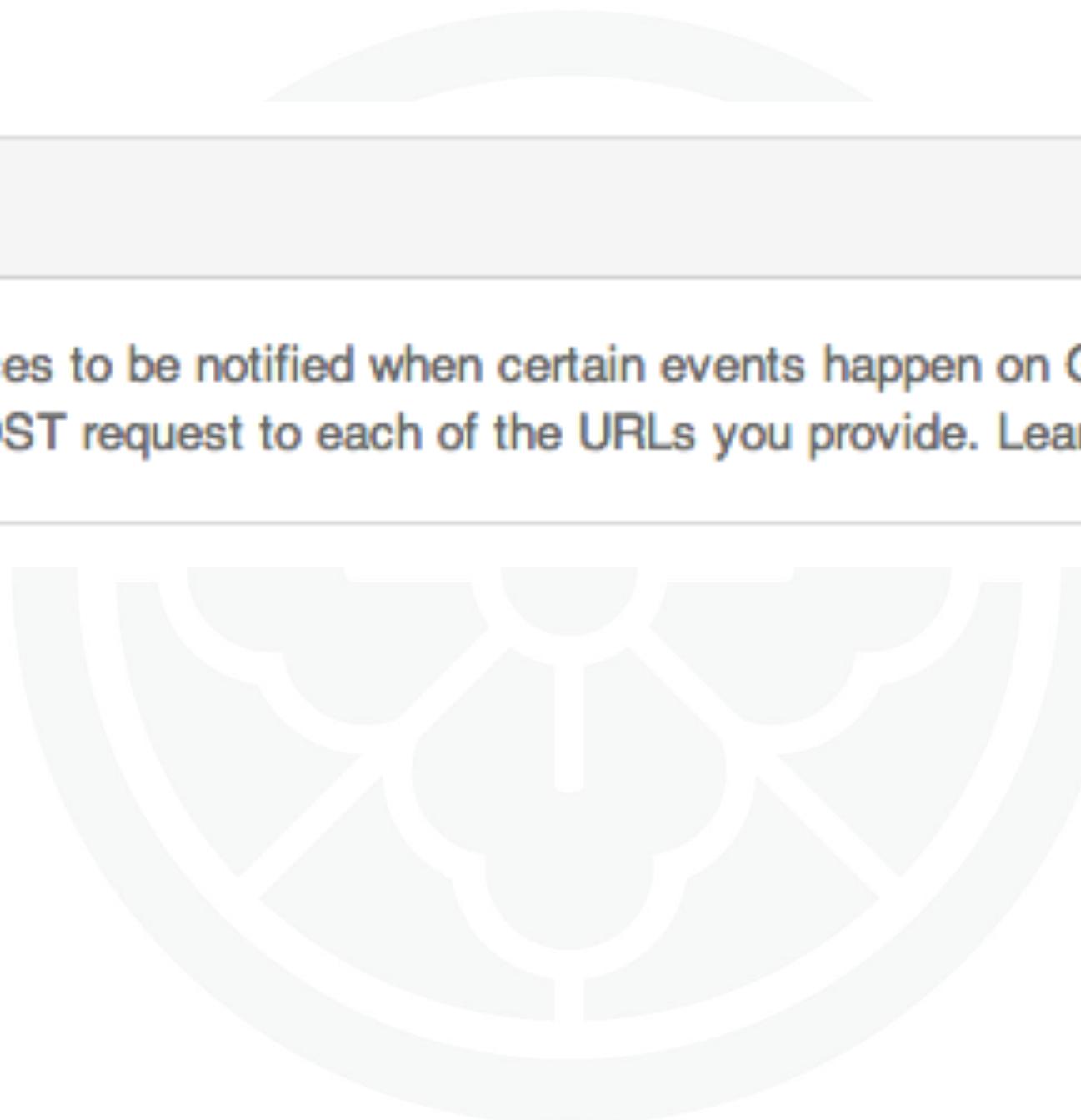
Webhooks allow external services to be notified when certain events happen on GitHub. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

Services

Add service ▾

Services are pre-built integrations that perform certain actions when events occur on GitHub. For more information on services check out our [Service Hooks Guide](#).

Implementing Custom Webhooks



Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen on GitHub. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

Adding a Webhook

Webhooks / [Add webhook](#)

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our developer documentation.

Payload URL

Content type

Secret

Which events would you like to trigger this webhook?

Just the push event.

Send me **everything**.

Let me select individual events.

Active
We will deliver event details when this hook is triggered.

Add webhook

The GitHub API

Level 6: Section 3

Working with the GitHub API

Simple curl commands show possibilities

- curl <https://api.github.com> - list of commands
- curl <https://api.github.com/emojis> - List of emojis!
- curl <https://api.github.com/users/peterbell> - Public info on user
- curl <https://api.github.com/users/peterbell/repos> - A users repos

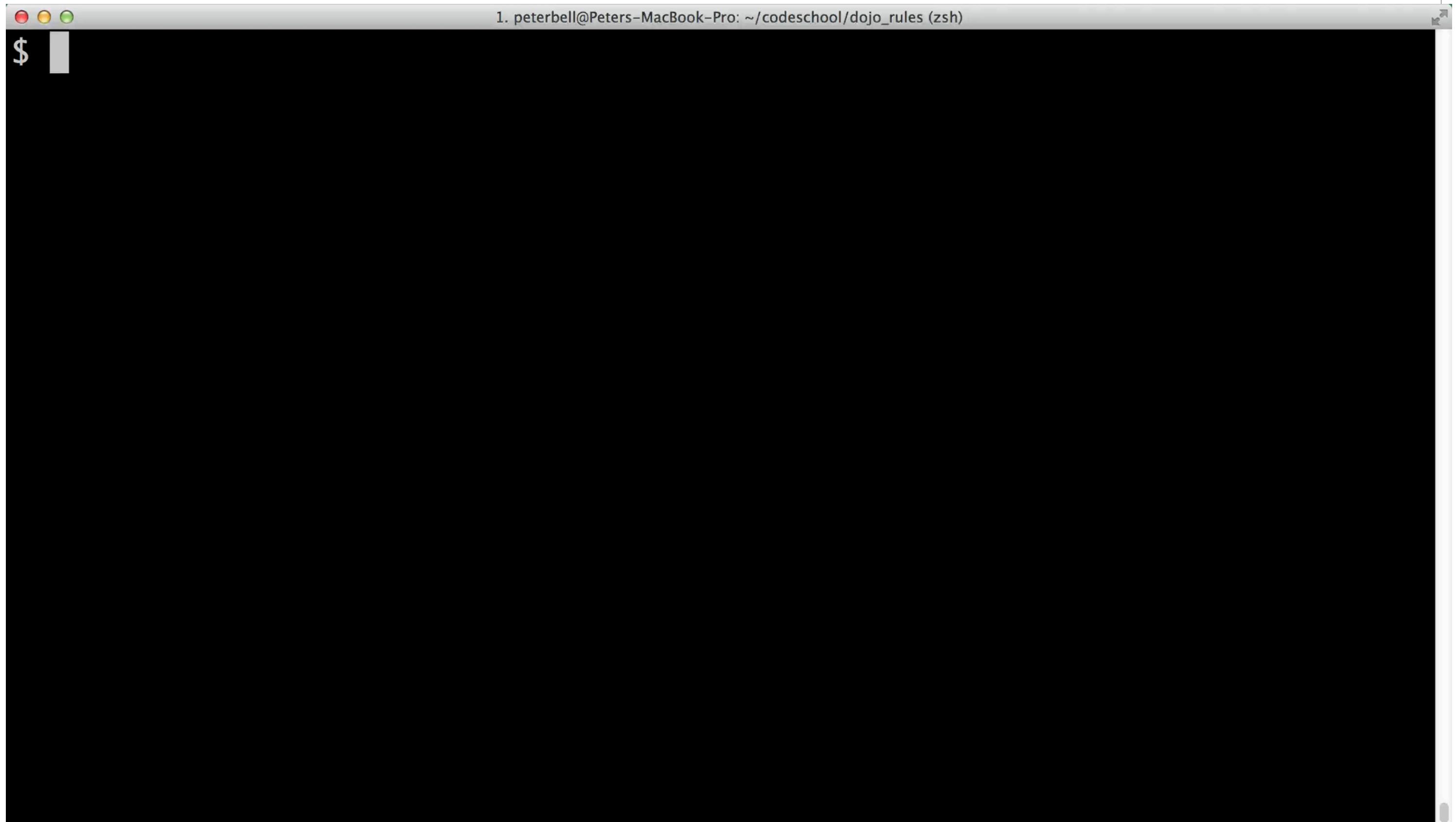
Authenticate:

- curl -i -H 'Authorization: token <your_authentication_token>'
<https://api.github.com/user>

Create a repo:

- curl -i -H 'Authorization: token <your_authentication_token>'
-d '{ "name": "test_repo" }' <https://api.github.com/user/repos>

Working with the GitHub API



1. peterbell@Peters-MacBook-Pro: ~/codeschool/dojo_rules (zsh)

```
$
```

Key API Capabilities Include Working With

- Activity
- Gists
- Git data
- Issues
- Organizations
- Pull Requests
- Repositories
- Search
- Users



• MASTERING • GITHUB



• MASTERING • GITHUB

Getting Started with Hub

Level 7: Section 1

Command Line GitHub



MAKES GIT BETTER WITH GITHUB

documentation: [man_hub](#)

hub is a command-line wrapper for git that makes you better at GitHub .

- Command line GitHub access
- Quicker, faster, mouse free!

Installing Hub

Mac

- \$ brew update
- \$ brew install hub

Linux/Windows

- \$ git clone https://github.com/github/hub.git
- \$ cd hub
- \$ rake install prefix=/usr/local

Using Hub

1. Use hub commands

- \$ hub clone
- \$ hub fork
- etc.

2. Alias hub as git

- \$ alias git=hub
- \$ git clone
- \$ git fork
- etc.

Notes:

- Don't alias in msysGit
- Tab completion:
 - http://go.codeschool.com/bash_completion
 - http://go.codeschool.com/zsh_completion

Creating a Repo Using Hub

```
$ git init new_repo  
$ cd new_repo  
$ vi README.md  
$ git add .  
$ git commit  
$ git create  
$ git push -u origin master  
$ git browse
```

Creating a Pull Request Using Hub

```
$ git checkout -b new_branch  
$ vi new_code.rb  
$ git add .  
$ git commit  
$ git push -u origin new_branch  
$ git pull-request
```

Cloning a Repo Using Hub

```
$ git clone repo_name
```

to clone one of your repo's

or

```
$ git clone username/repo_name
```

to clone someone else's

Forking a Repo Using Hub

Forking a repo:

```
$ git clone deadlyvipers/dojo_rules  
$ cd dojo_rules  
$ git fork
```

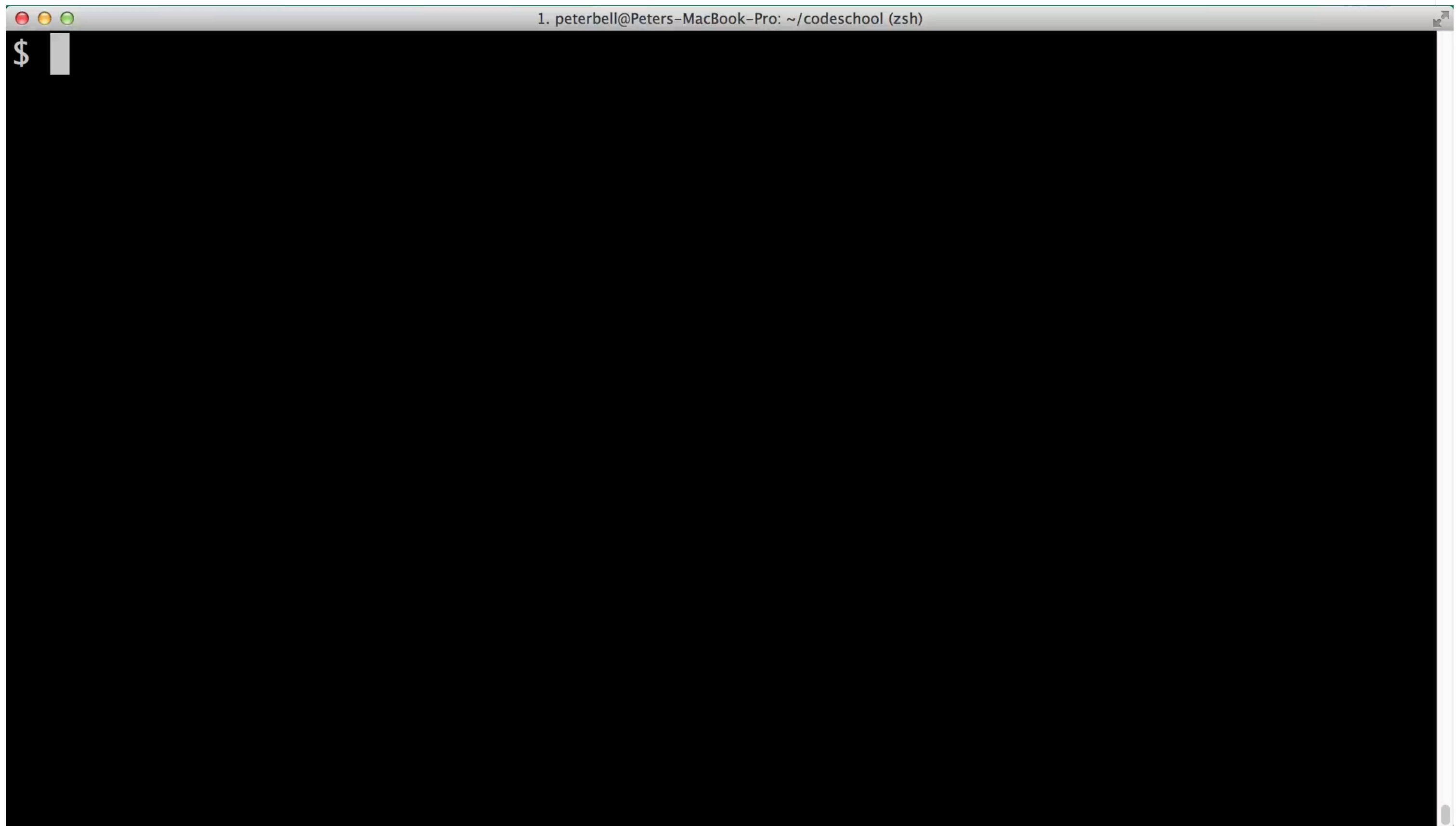
Just need to specify repos/branches to use pull-request.

```
$ git checkout -b new_branch  
$ vi new_code.rb  
$ git add .  
$ git commit  
$ git pull-request -b peterbell:master  
  -h peterbell:new_branch  
$ git browse dojo_rules
```

specifying PR base (-b) and head (-h)

defaults to <username>/dojo_rules

Hub Basics



1. peterbell@Peters-MacBook-Pro: ~/codeschool (zsh)

```
$
```

Collaborating with Hub

Pulling from Forks Using Hub

```
$ git remote add username
```

or

```
$ git fetch username
```

Cherry-picking from Forks Using Hub

```
$ git cherry-pick username@SHA
```

Checking out a Pull Request from a Fork

```
$ git checkout https://github.com/username/repo/pull/1 custom-branch-name
```

Merging a Pull Request from a Fork

```
$ git merge https://github.com/username/repo/pull/1
```

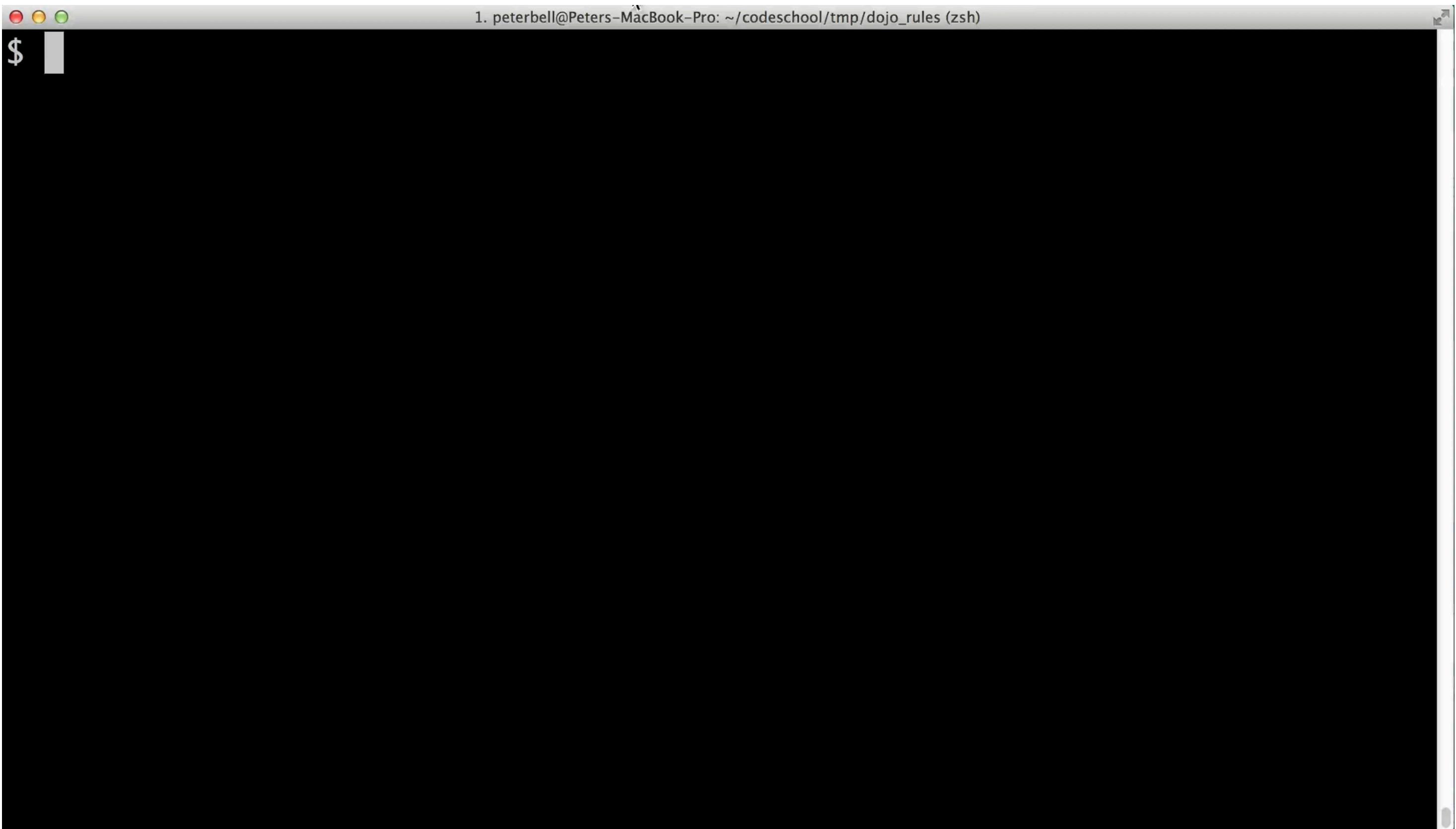
If it's good, then push the merge.

```
$ git push
```

Otherwise you can blow away the merge commit.

```
$ git reset --hard HEAD~1
```

Collaborating with Hub



1. peterbell@Peters-MacBook-Pro: ~/codeschool/tmp/dojo_rules (zsh)

```
$
```

• MASTERING • GITHUB