



git + **GitHub**



git

What is Git?

Git

is a distributed revision control system with an emphasis on speed, data integrity, and support for distributed, non-linear workflows.



git Git History

- Came out of Linux development community
- Linus Torvalds, 2005
- Initial goals:
 - Speed
 - Support for non-linear development (thousands of parallel branches)
 - Fully distributed
 - Able to handle large projects like Linux efficiently



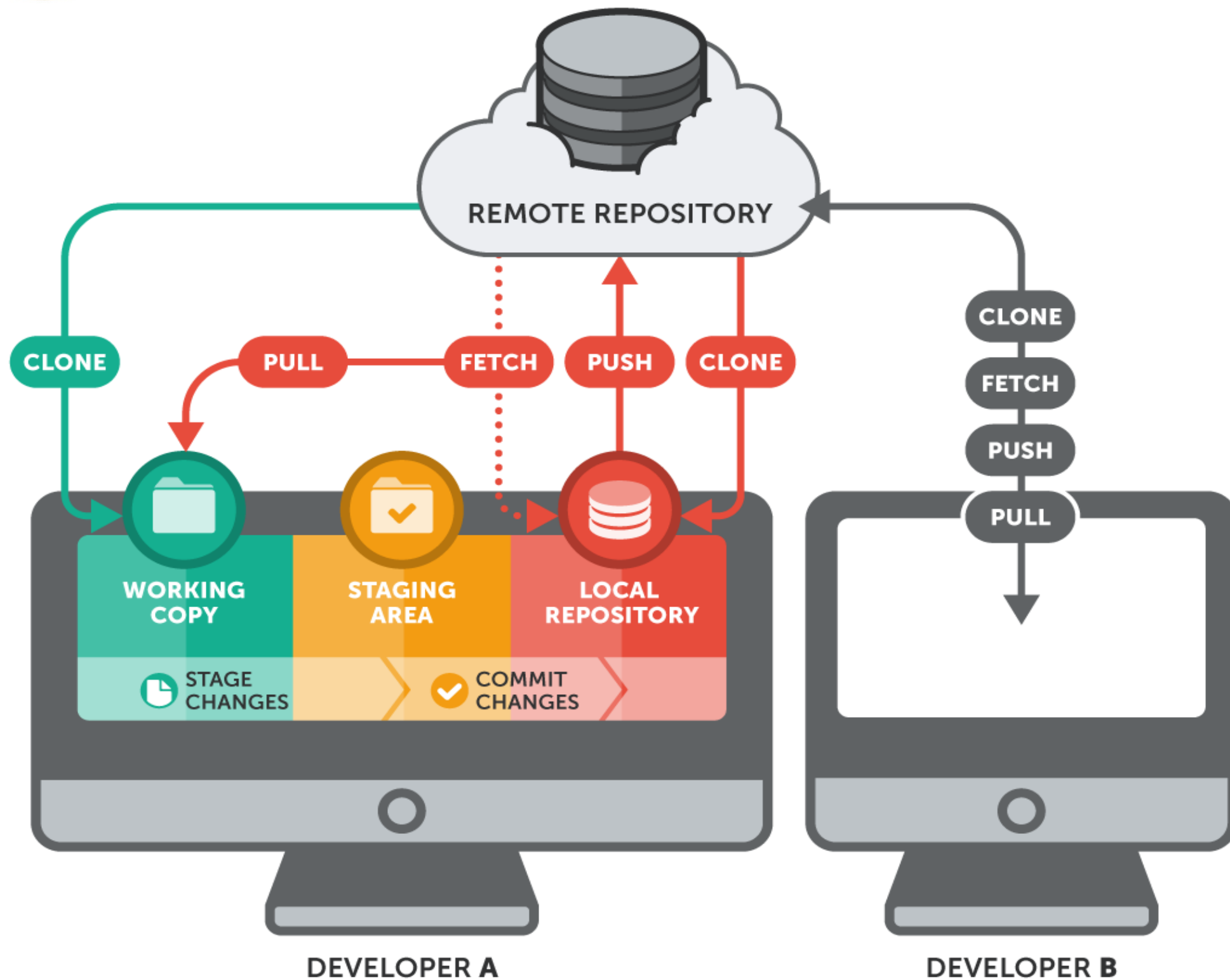
git

Why Git?

- ★ Save Time - Git is lightning fast!
- ★ Work Offline - With Git, almost everything is possible simply on your local machine
- ★ Undo Mistakes - There's a little "undo" command for almost every situation.
- ★ Don't Worry - Losing data or breaking a repository beyond repair is really hard to do
- ★ Make useful commits - You can determine exactly which changes shall be included in your next commits
- ★ Work in your own way - When working with Git you can use your very own workflow
- ★ Don't Mix Things Up - Git branching is fast & easy!
- ★ Go With The Flow - Ruby On Rails, jQuery, Perl, the Linux Kernel, a lot of Microsoft products



Basic Git Commands





Create new repository

Create a new directory, open it and perform a

```
# git init
```

to create a new git repository.



Checkout a repository

Create a working copy of a local repository by running the command

```
# git clone /path/to/repository
```

when using a remote server, your command will be

```
# git clone username@host:/path/to/repository
```




Check status

List the files you've changed and those you still need to add or commit:

git status

Show content differences

git diff

Lists version history

git log



Add & Commit

You can propose changes (add it to the **Index**) using

```
# git add <filename>
```

```
# git add *
```

This is the first step in the basic git workflow. To actually commit these changes
use

```
# git commit -m "Commit message"
```

Now the file is committed to the **HEAD**, but not in your remote repository yet.



Pushing changes

To send those changes to your remote repository, execute

```
# git push origin master
```

Change ***master*** to whatever branch you want to push your changes to.

If you have not cloned an existing repository and want to connect your repository to a remote server, you need to add it with

```
# git remote add origin <server>
```

Now you are able to push your changes to the selected remote server



Branching

To create a new branch named "feature_x" and switch to it using

```
# git checkout -b feature_x
```

switch back to master

```
# git checkout master
```

and delete the branch again

```
# git branch -d feature_x
```

a branch is *not available to others* unless you push the branch to your remote repository

```
# git push origin <branch>
```



Update & Merge

To update your local repository to the newest commit, execute

```
# git pull
```

in your working directory to *fetch* and *merge* remote changes.

to merge another branch into your active branch (e.g. master), use

```
# git merge <branch>
```

in both cases git tries to auto-merge changes. Unfortunately, this is not always possible and results in *conflicts*. You are responsible to merge those *conflicts* manually by editing the files shown by git or use a mergetool to handle the conflicts.



Undoing Things

To amend last commit you can use

```
# git commit --amend
```

If you want to reset the working directory to a previous commit

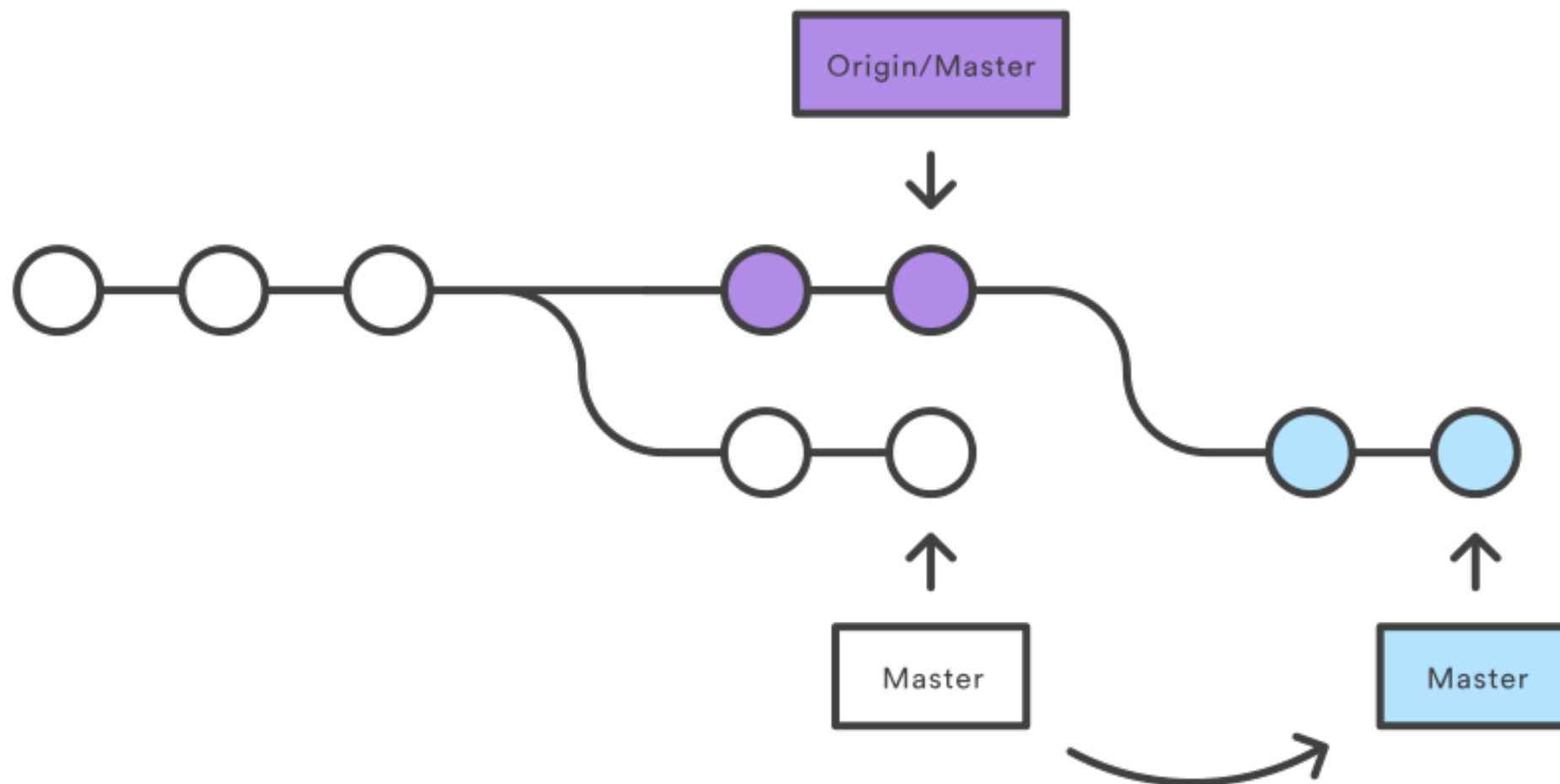
```
# git reset [--hard] <commit>
```



Git Workflows

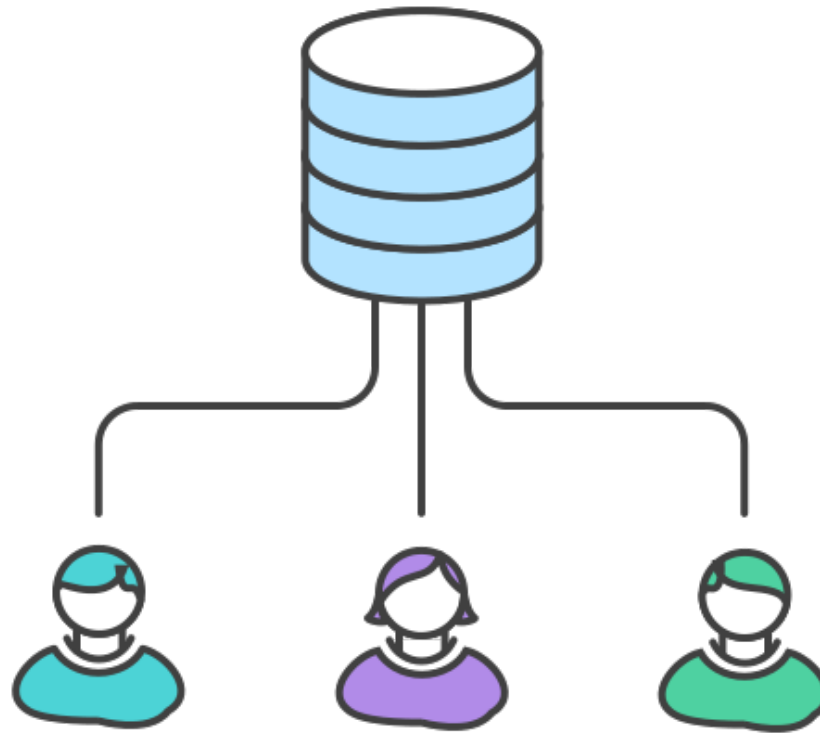


Centralized Workflow

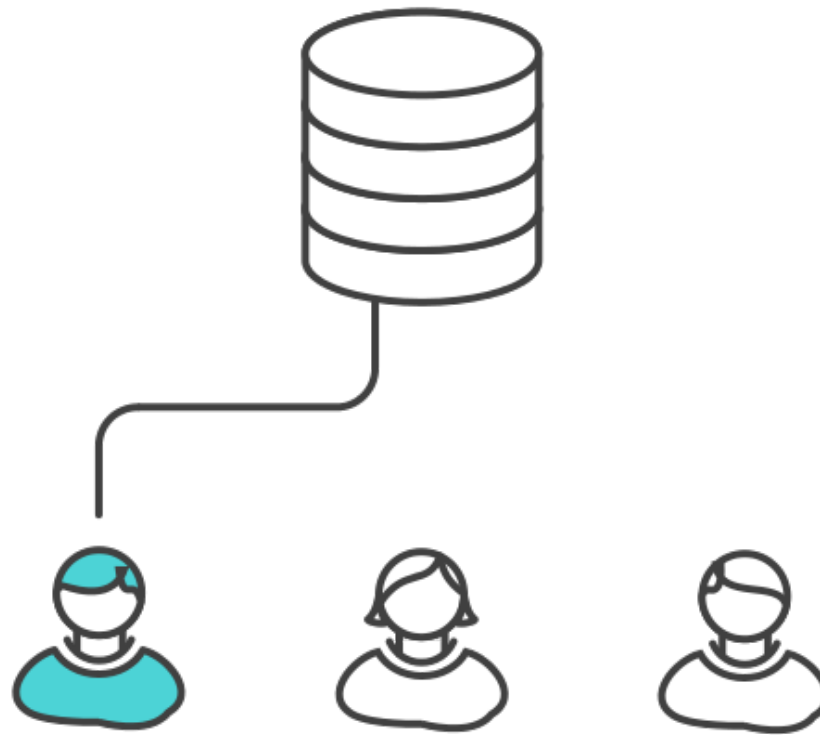




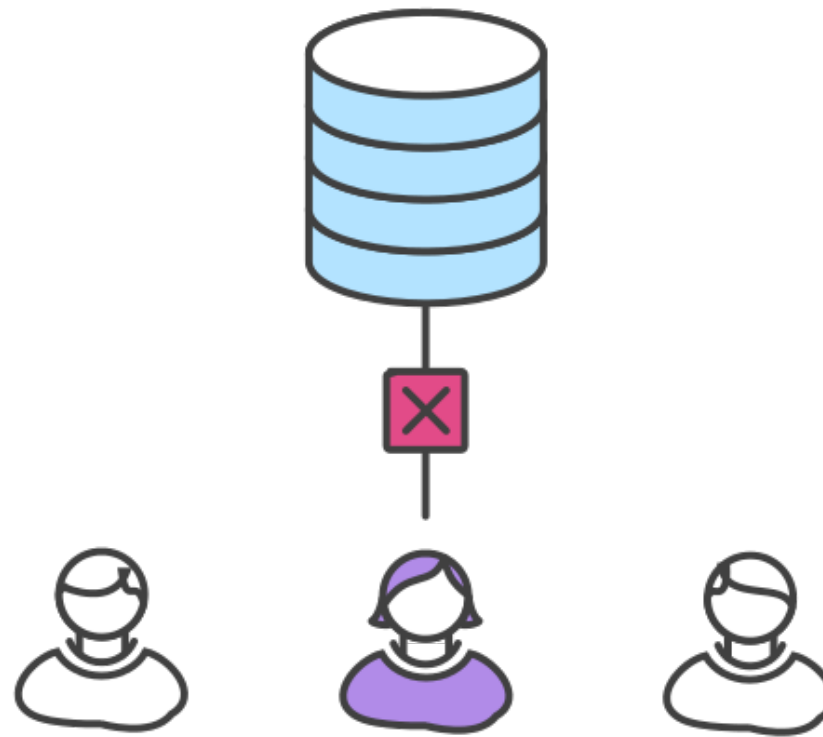
How it works?



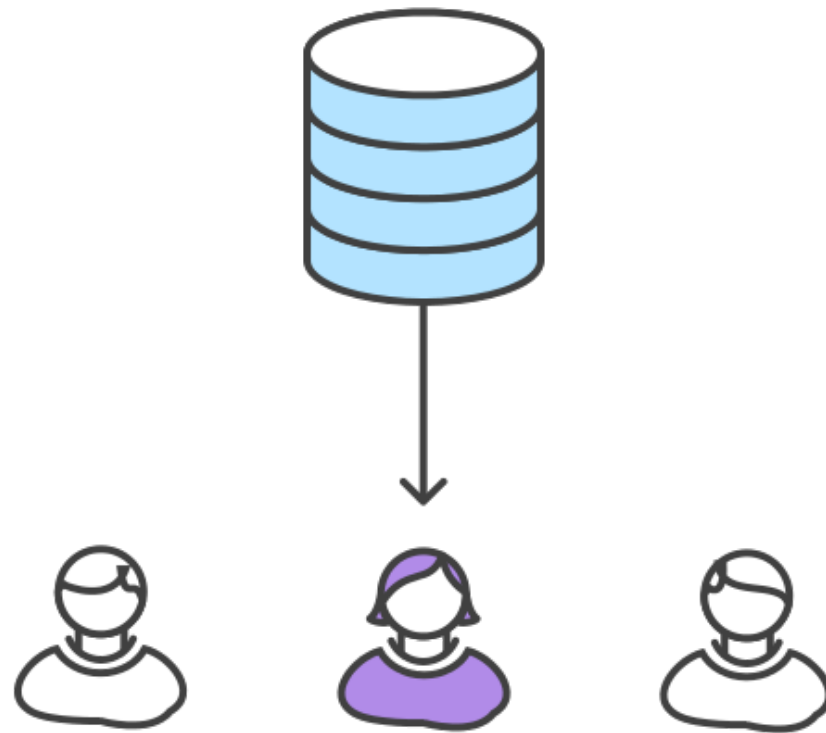
All team members clone the repository



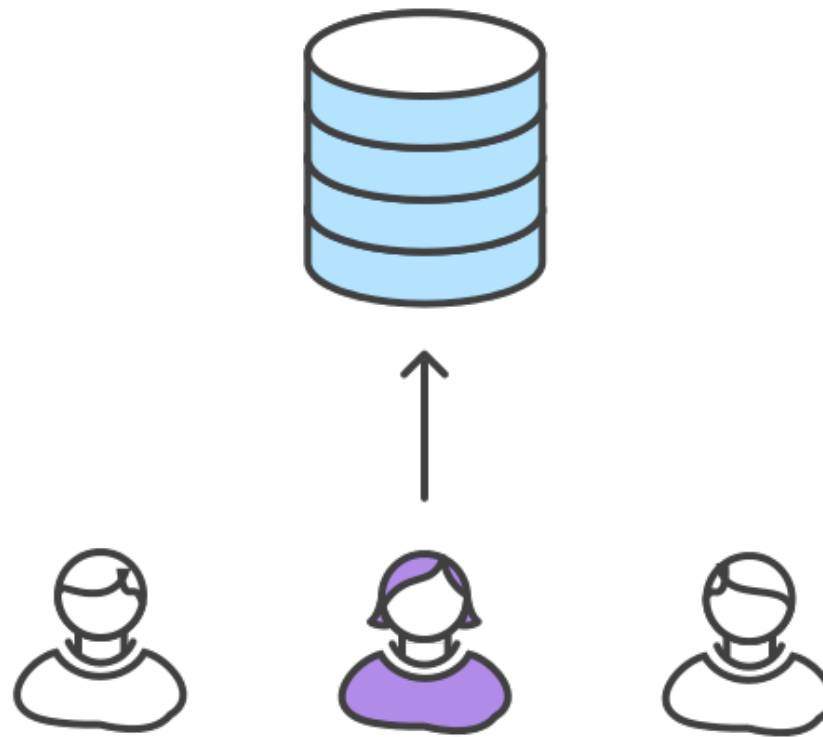
John makes
a commit(s) and push it(them) to the repository



Jane tries to push her commit



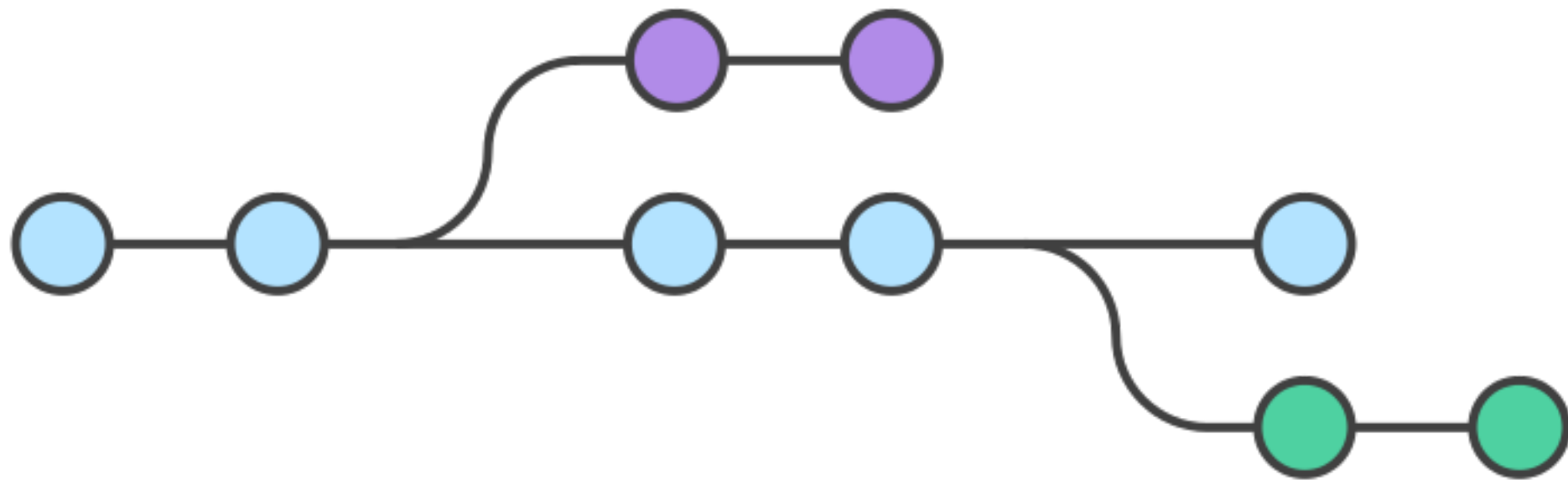
Jane rebase on top of John commit(s)
and resolve any merge conflicts



Jane successfully push her commit(s)

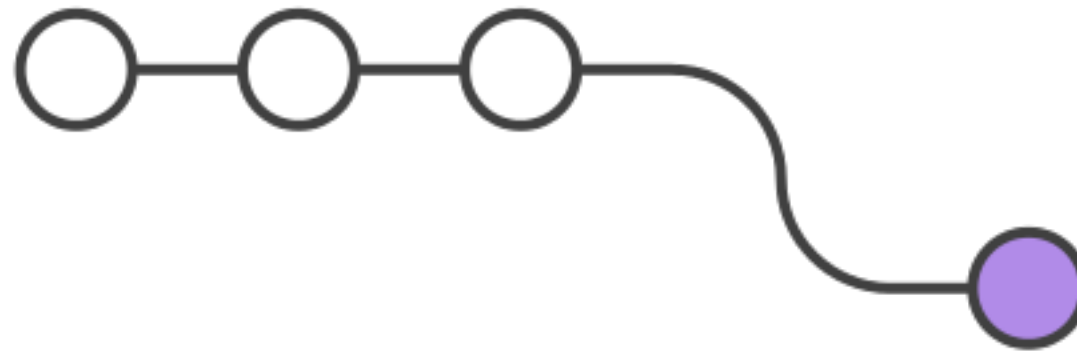


Feature Branch Workflow

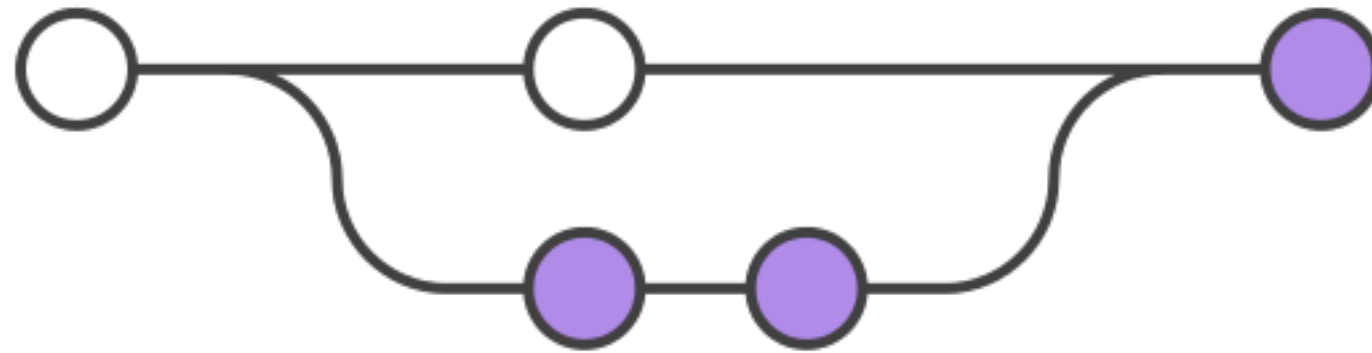




How it works?



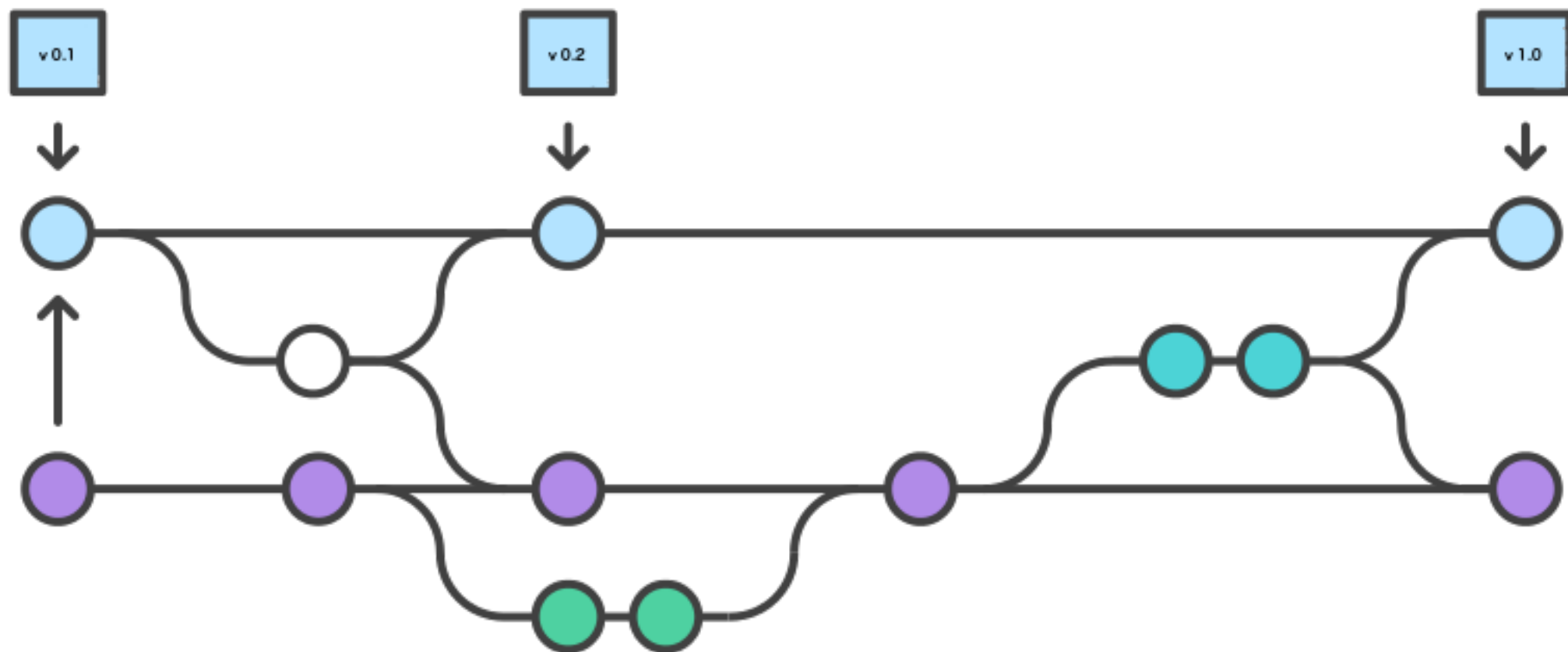
John begins new feature branch based on **master**



John completes his feature
and merge (or files a pull request)
his branch to **master**



Gitflow Workflow

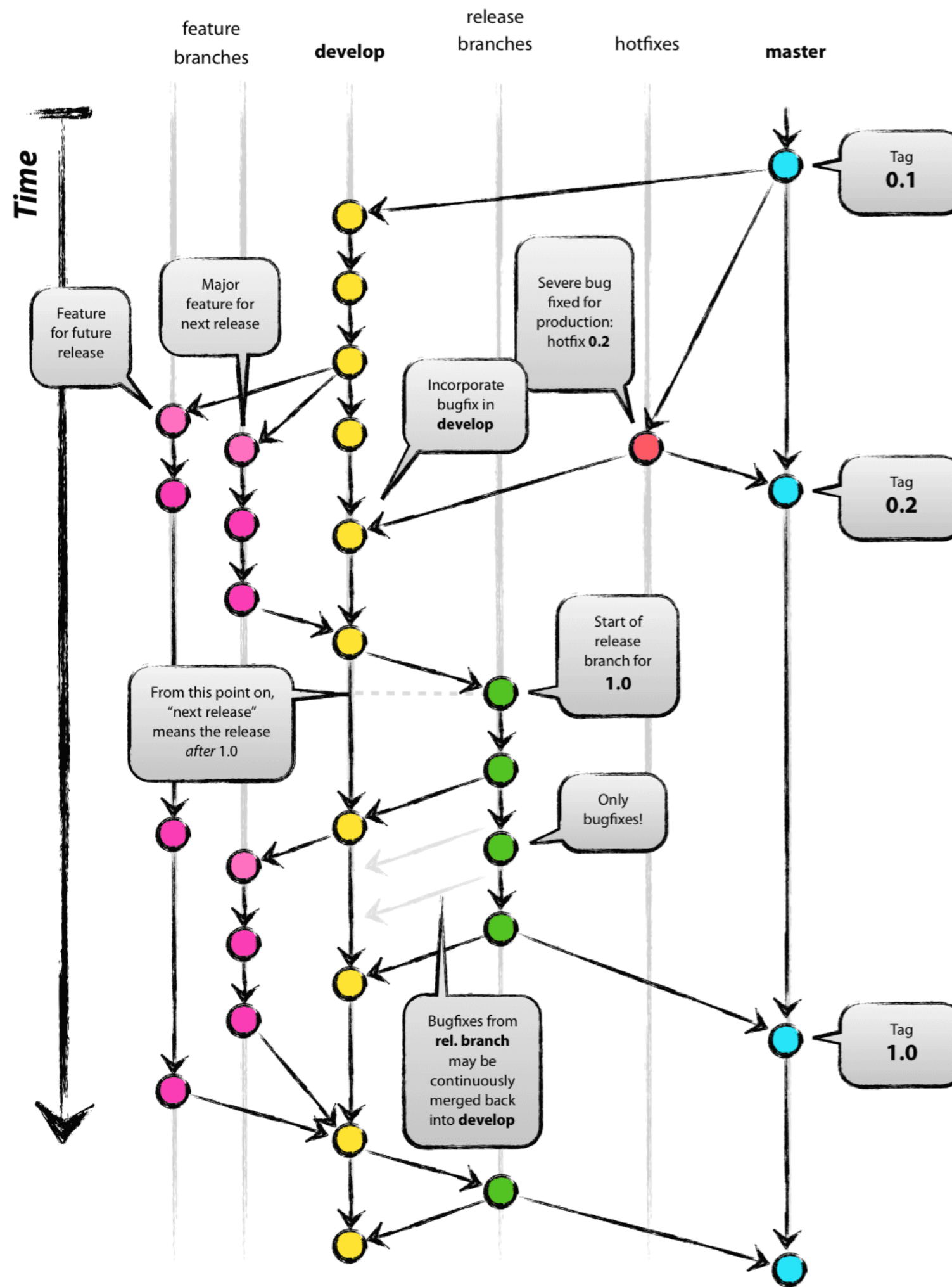




How it works?

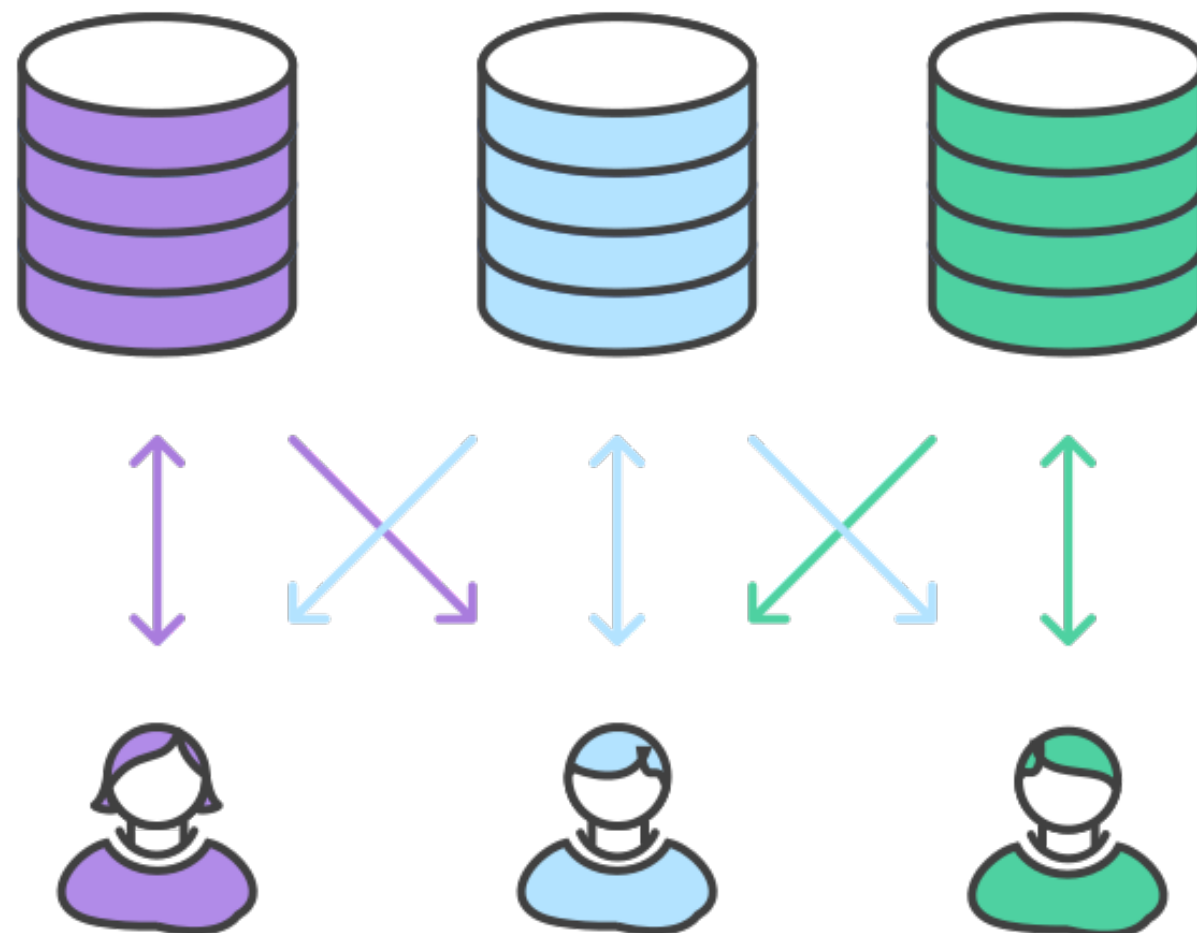


git





Forking Workflow

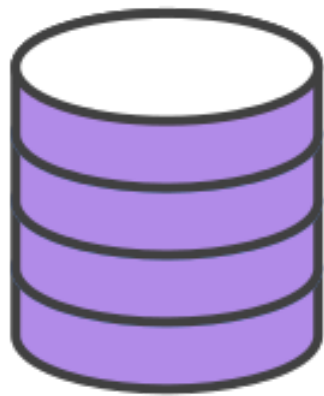




How it works?

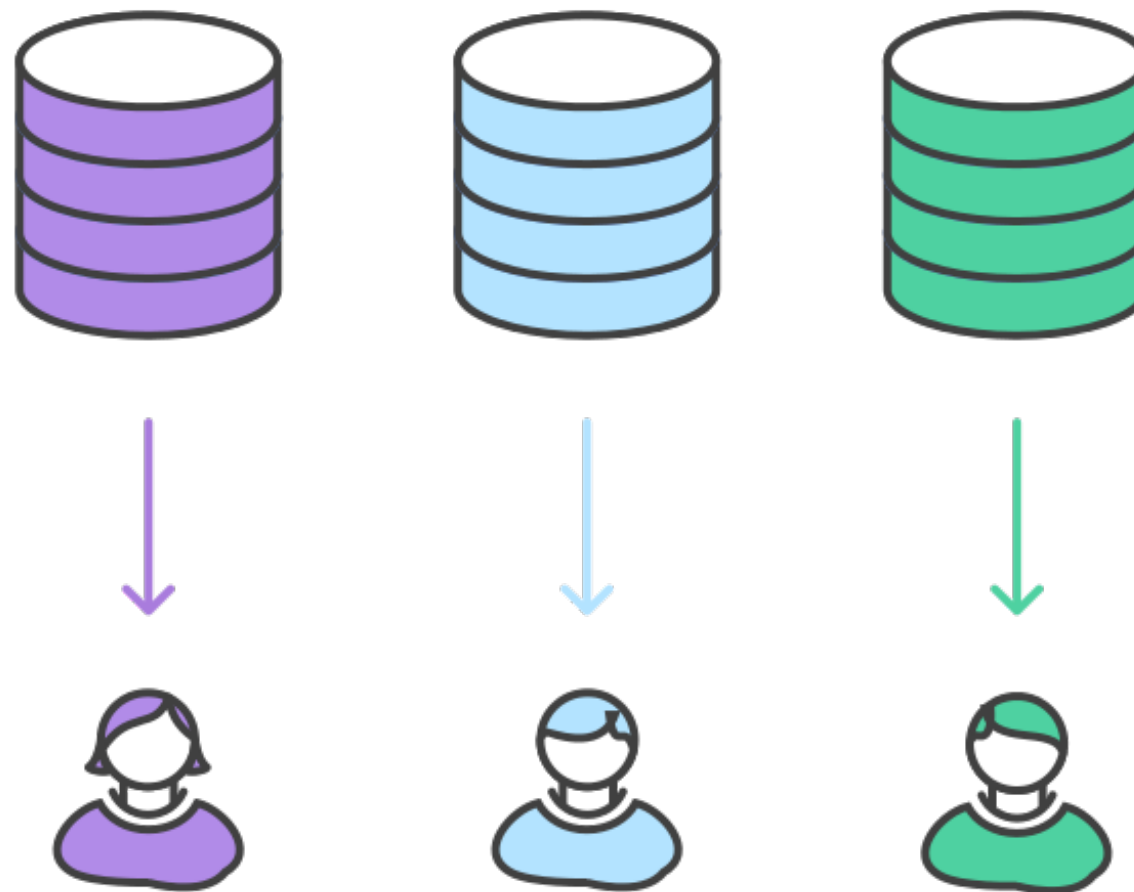


Pick a “blessed” upstream repository
that you want to fork



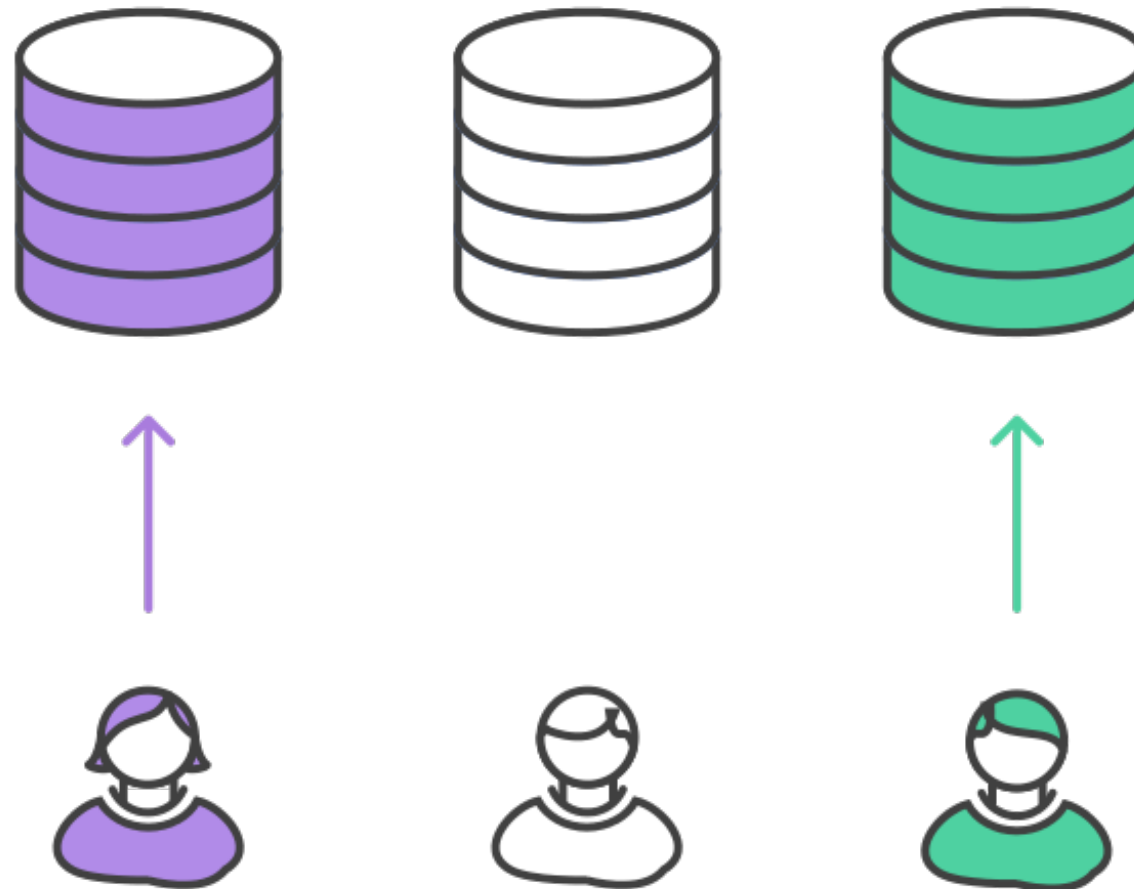
Team member(s) fork* this repository

* Fork is just a server-side clone



- All members get a local clone of the forked repository
- Additionally add the original repository as upstream, in order to sync any feature changes

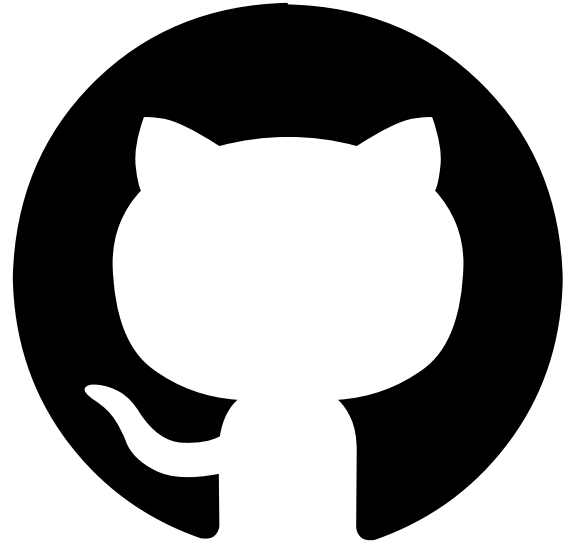
```
# git remote add upstream <repo>
```



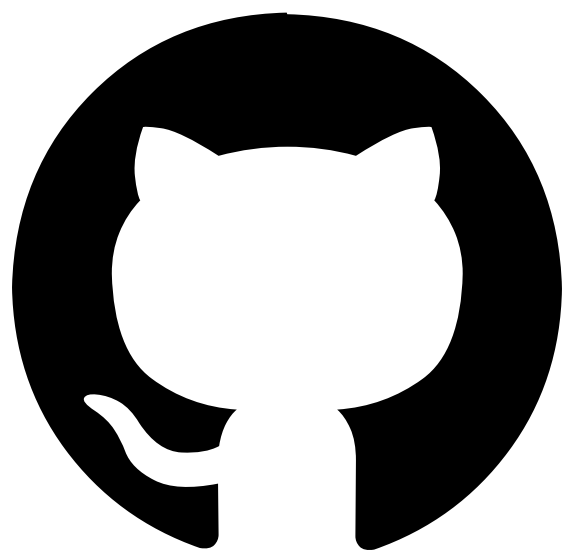
You can use any workflow
to implement features



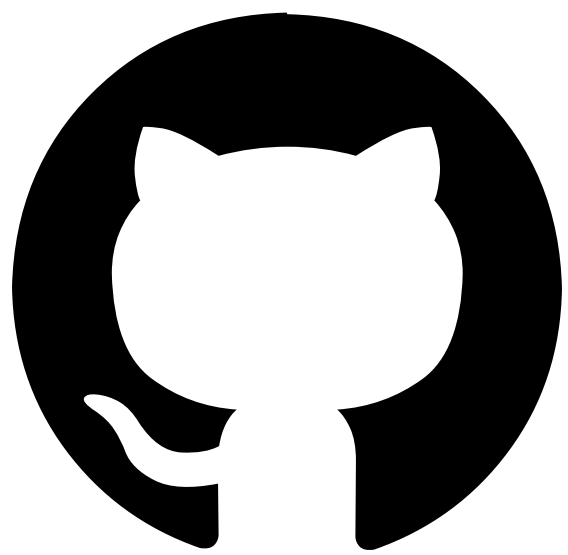
When you are ready, a team member,
should file a pull request
to the owner of the original repository



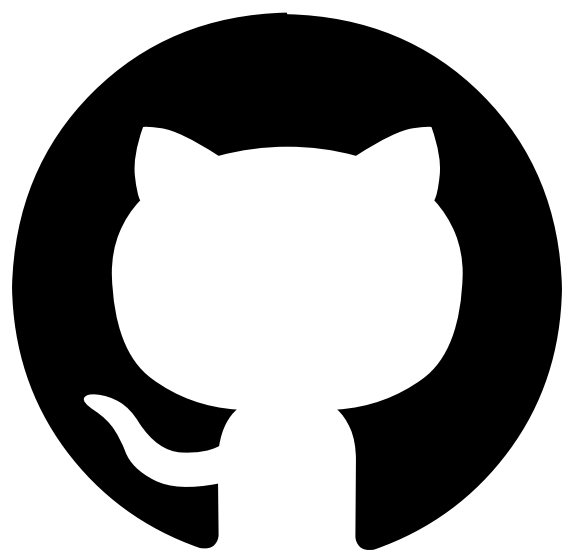
GitHub



Features



Integrated issue
tracking



Issue Listings

PUBLIC  rails / rails

 Watch ▾

1,524

 Star

22,519

 Fork

8,425

Issues

Pull requests

Labels

Milestones

Filters ▾

 is:open is:issue

New issue

 266 Open ✓ 5,567 Closed


Author ▾

Labels ▾

Milestones ▾


Assignee ▾

Sort ▾

 Migration change_column array:true, default: '{}' don't work rails 4.1.4


#16233 opened 2 hours ago by nilocoelhojunior

 0

 'becomes' method does not copy unsaved attributes **activerecord**

#16206 opened 3 days ago by thehappycoder

 4

 ParamsWrapper, nested ressource, parent id never included **actionpack**


#16194 opened 3 days ago by elbartto132

 0

 ActiveRecord::Relation .size regression with distinct select **activerecord** **regression**


#16182 opened 4 days ago by aripollak

 0

 ActionMailer: calling "attachments[]" after "mail" leads to unexpected behavior **actionmailer**


#16163 opened 6 days ago by joakimk

 0

 ActionController::UrlFor requires more as a Rack-compatible request

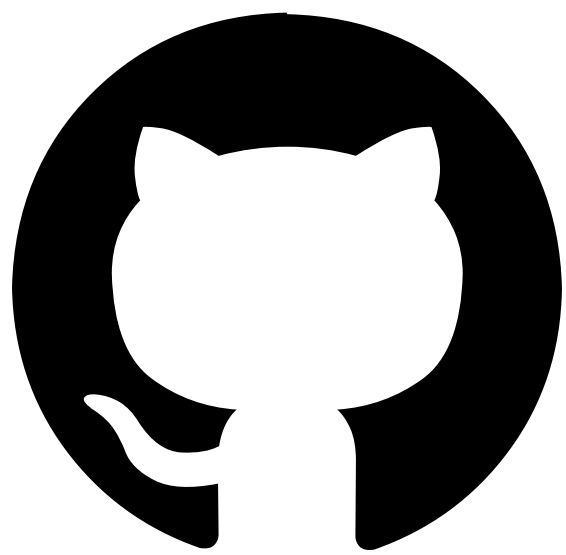
#16160 opened 6 days ago by dspaeth-faber

 0

 Counter cache doesn't update if parent model primary key changes

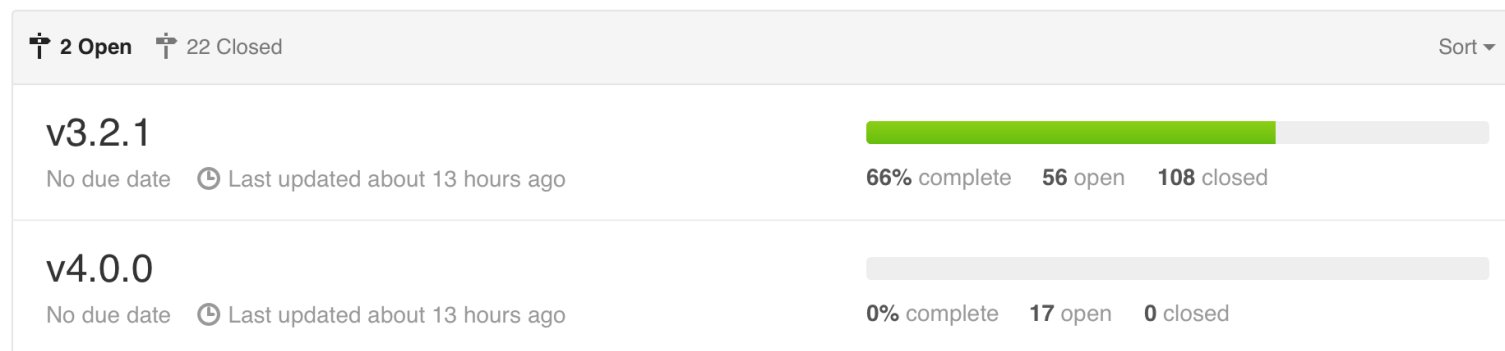
#16159 opened 6 days ago by noinkling

 1

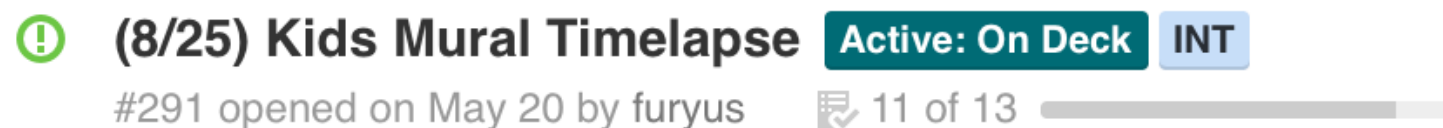


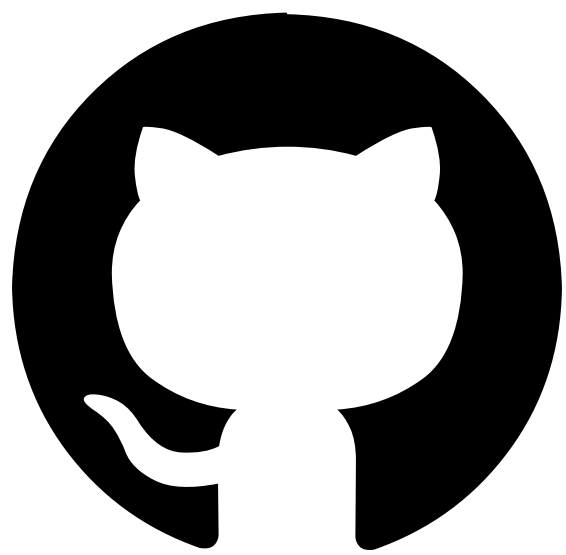
Milestones & Labels

Milestones are great at helping everyone work towards a goal. Set a due date, name your milestone, then start grouping issues together.



Labels are another way to organize your issues and can be customized with your own colors.





Commit keywords

Reference commits and other issues

Issues know all about commits and other issues.

Type in a commit SHA or an issue number and it will automatically be linked.

Menu items disappear after refreshing #10989

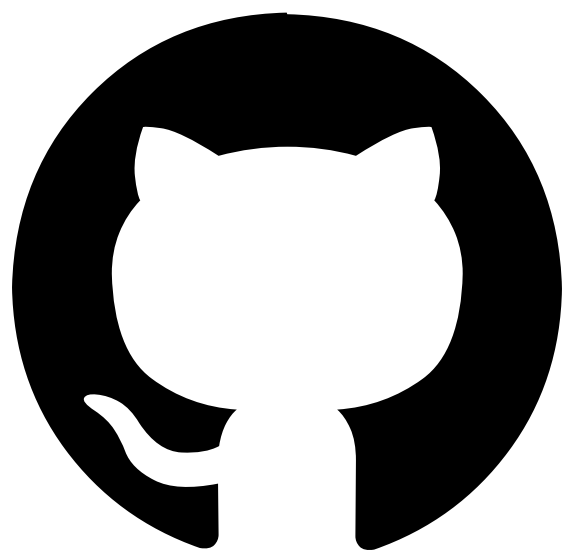
🔔 Open

holman opened this issue just now · 0 comments

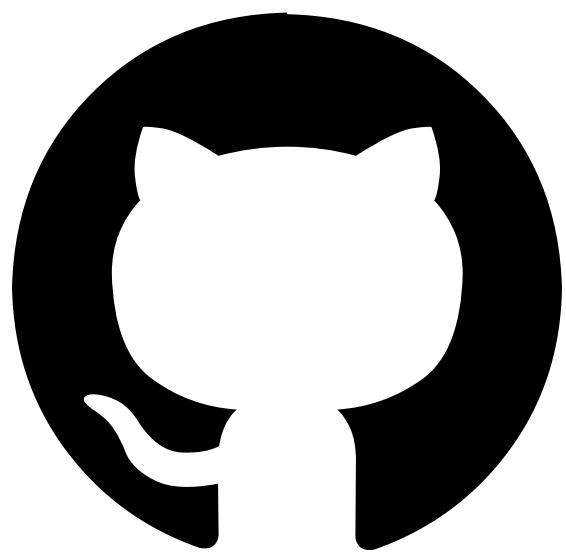


holman commented just now

This was fixed in [#1636](#), but `a0d8d56` broke it. @mdo do you have any thoughts?



Collaborative code review



Pull Requests

Pull Requests are **living conversations** that streamline the process of discussing, reviewing, and managing changes to code.

Pull Request = Code + Issue + Code Comments

Make left/top auto value consistent across browsers #1241

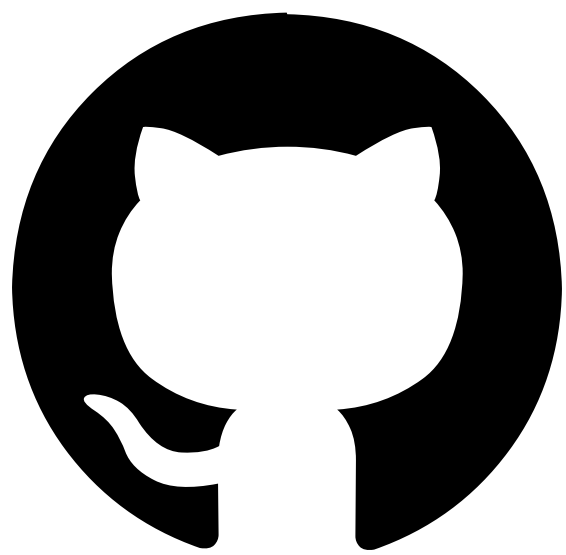
[Open](#) yiminghe wants to merge 8 commits into `jquery:master` from `yiminghe:bug_13767`

Conversation 23 Commits 8 Files changed 3 +27 -6

Showing 3 changed files with 27 additions and 6 deletions. Unified Split

13 src/css.js View

		@@ -506,9 +506,16 @@ jQuery(function() {
506	506	jQuery.cssHooks[prop] = {
507	507	get: function(elem, computed) {
508	508	if (computed) {
509	-	computed = curCSS(elem, prop);
510	-	// if curCSS returns percentage, fallback to offset
511	-	return rnumnonpx.test(computed) ?
	509	var isAutoPosition,
	510	elStyles = getStyles(elem),
	511	position = curCSS(elem, "position", elStyles);
	512	computed = curCSS(elem, prop, elStyles);
	513	isAutoPosition = computed === "auto";
	514	if (isAutoPosition && position === "relative") {
	515	return "0px";
	516	}
	517	// if curCSS returns percentage or auto, fallback to offset
	518	return isAutoPosition && position !== "static" rnumnonpx.test(computed) ?
512	519	jQuery(elem).position()[prop] + "px" :
513	520	computed;
514	521	}



Commit Comments

You can have conversations on entire commits as well as individual lines of code.

90	90	// Use window.getComputedStyle bec
91	91	if (window.getComputedStyle) {
92		- support.pixelPosition = (window.getComputedSt
93		- support.boxSizingReliable = (window.getComput
	92	+ divStyle = window.getComputedStyle(div, null

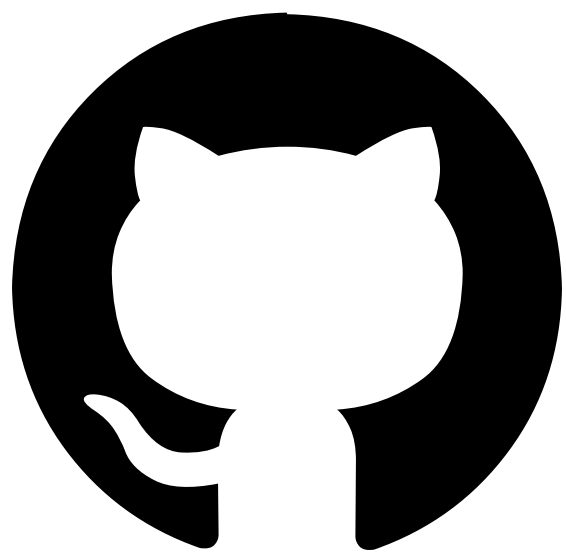


mzgol added a note on Apr 15, 2013

This adds 5 bytes but prevents `getComputedStyle` to be invoked twice
opinion?

Add a line note

	93	+ support.pixelPosition = divStyle.top !== "1%"
	94	+ support.boxSizingReliable = divStyle.width ===



Compare View

With GitHub you can easily and efficiently **compare any two branches** in your project or network.

master@{1day} ... master

15 commits

11 files changed

0 commit comments

10 contributors

Commits on Dec 03, 2014

GBH fixing mailer previews for apps with globbing route ccc3ddb

Commits on Dec 06, 2014

kirs Test coverage for ActiveJob generator 1bf42df

Showing 11 changed files with 85 additions and 17 deletions.

Unified Split

2 actionmailer/lib/action_mailer/railtie.rb

@@ -41,7 +41,7 @@ class Railtie < Rails::Railtie # :nodoc:

41 41 options.each { |k,v| send("#{k}=", v) }

42 42

43 43 if options.show_previews

44 - app.routes.append do

44 + app.routes.prepend do

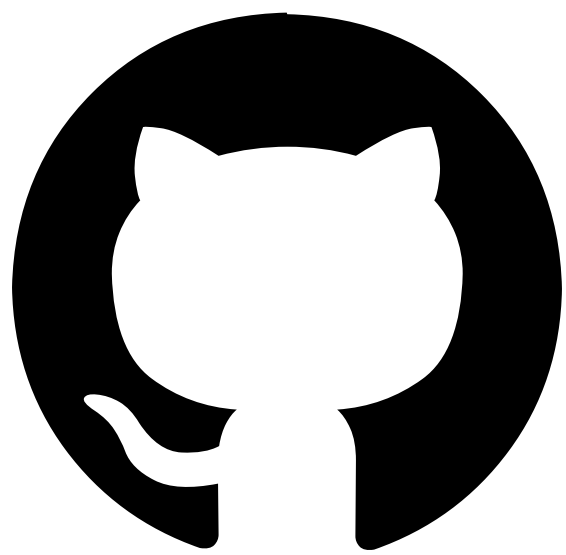
45 45 get '/rails/mailers' => "rails/mailers#index"

46 46 get '/rails/mailers/*path' => "rails/mailers#preview"

47 47 end

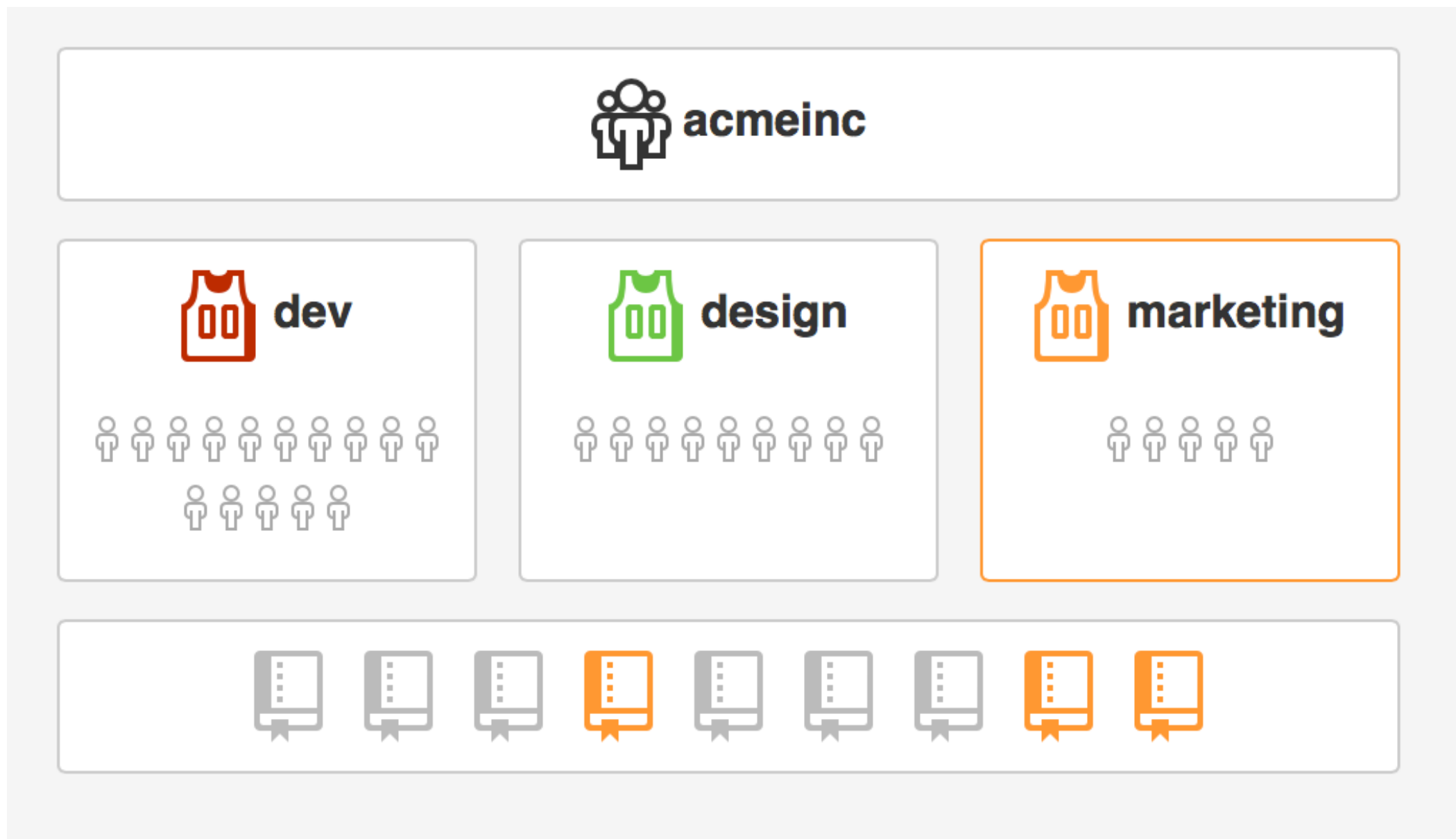
12 actionpack/lib/action_dispatch/routing/mapper.rb

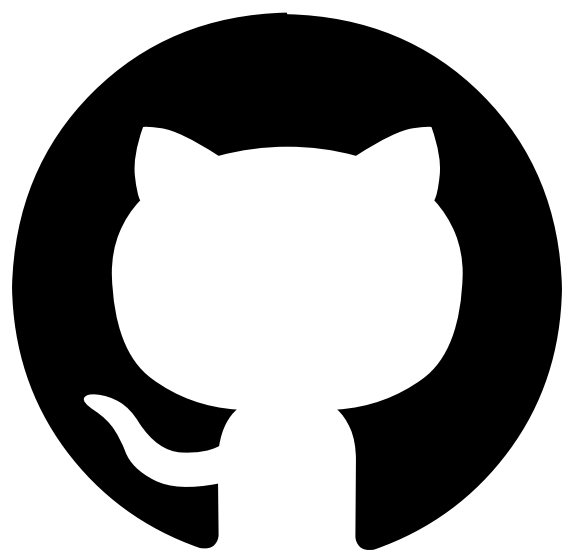
@@ -579,14 +579,15 @@ def mount(app, options = nil)



Easily manage teams within organizations

Members can be granted **read**, **read-write**, or **admin-level** access to repositories.





Text entry with understated power

GitHub rely on **GitHub Flavored Markdown** for formatting text

The **autocompleter** makes quick work of mentioning people and teams, creating links to other issues, and adding the perfect Emoji.

I think @Haacked fixed this in #1

#15 WTFPL

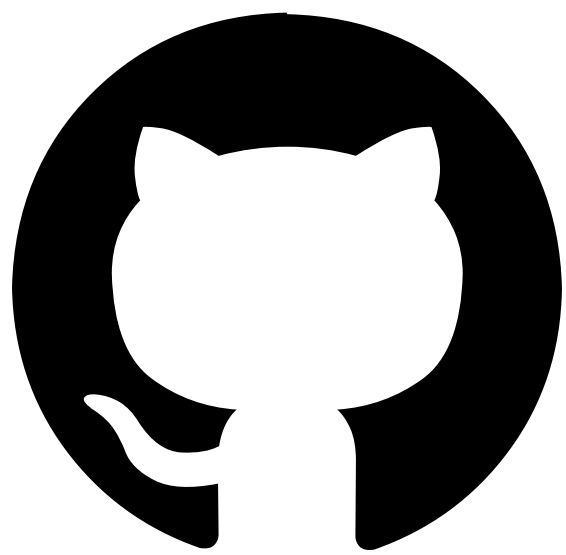
#17 **The WTFPL, also a choice**

#1 **MIT and BSD also allow private use**

#13 **Add links to additional resources**

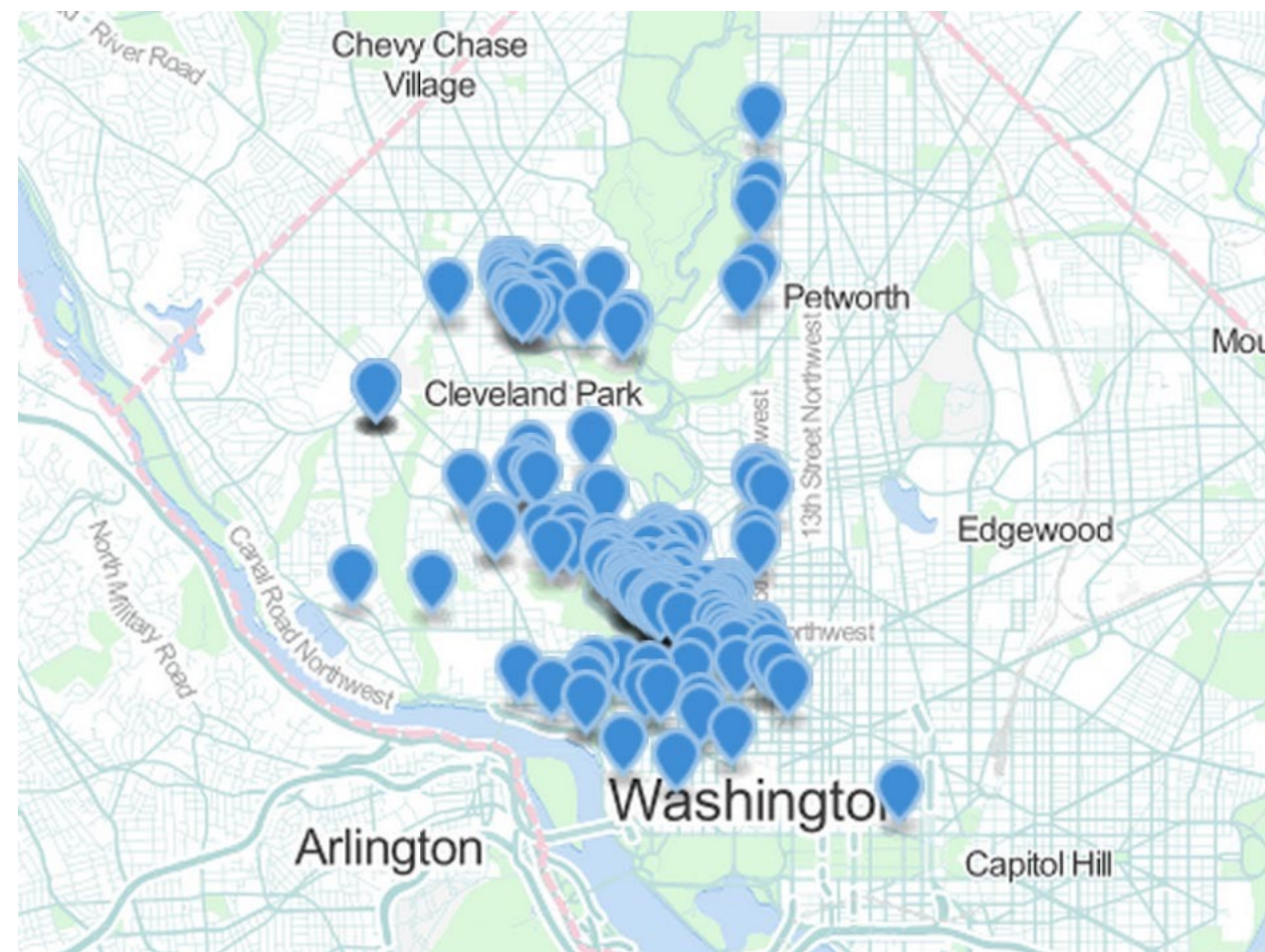
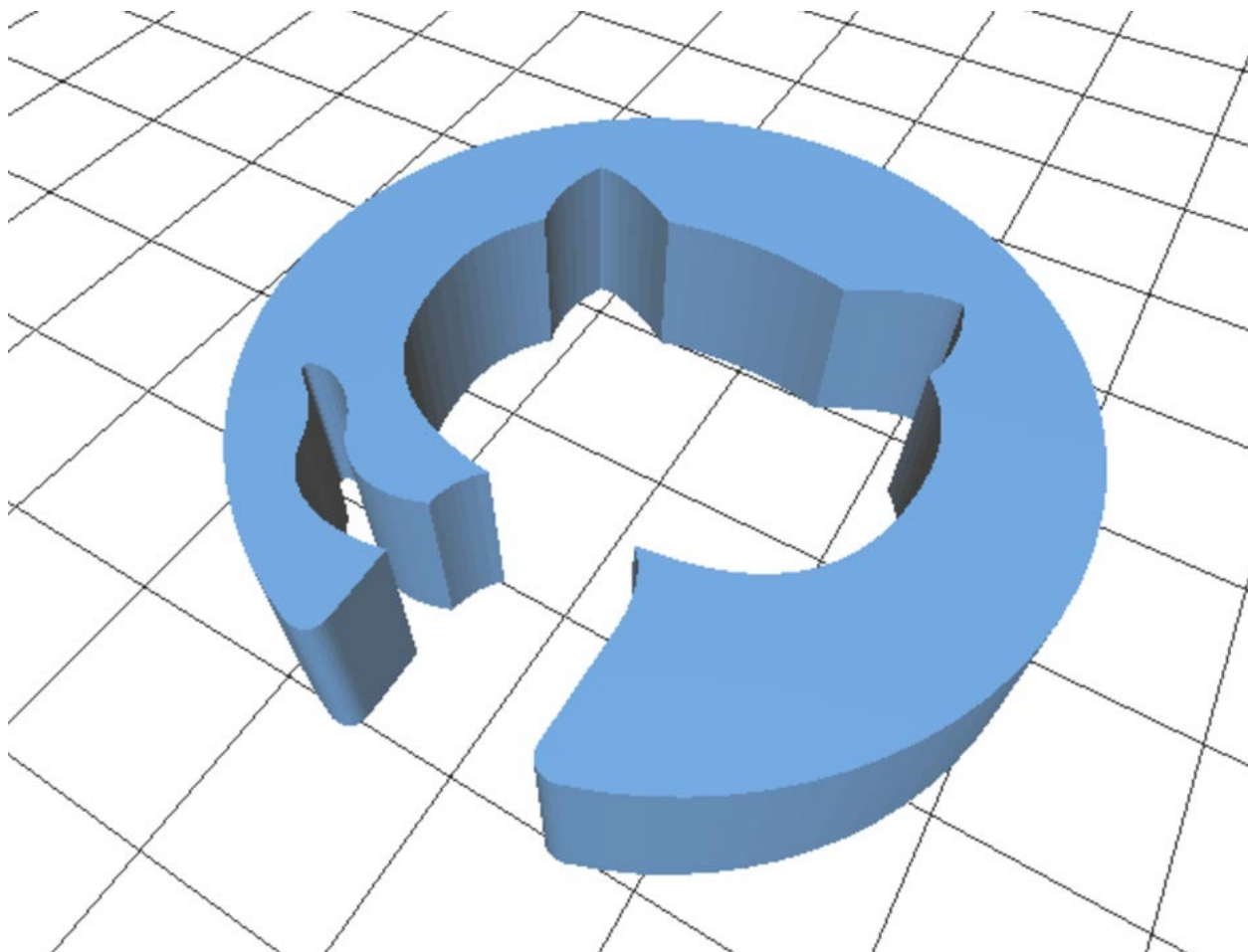
#12 **Remove implicit patent grant claim**

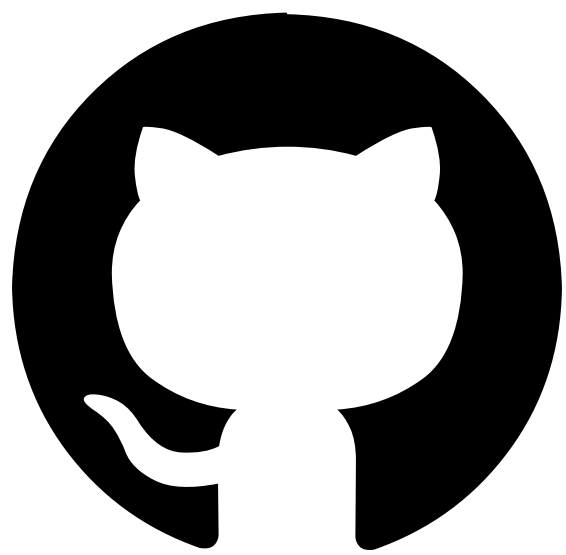
Attach images by dragging & dropping, [selecting them](#), or pasting from the clipboard.



Syntax highlighted code & rendered data

Syntax highlighting currently supports **over 200 programming languages**.
GitHub can render data formats like STL 3D models, CSV files, and GeoJSON maps in the browser.

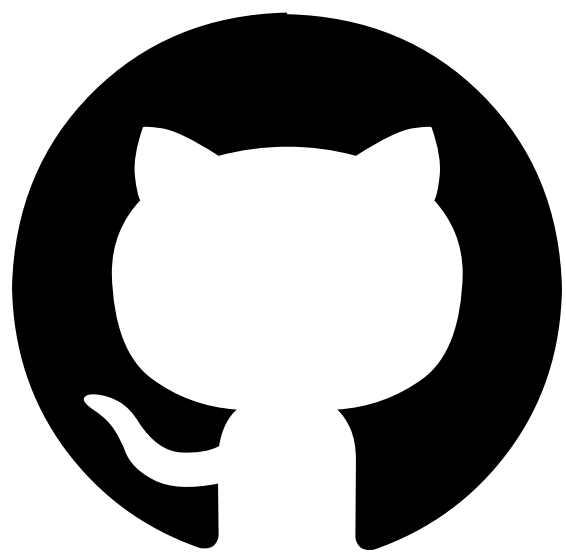




Robust API

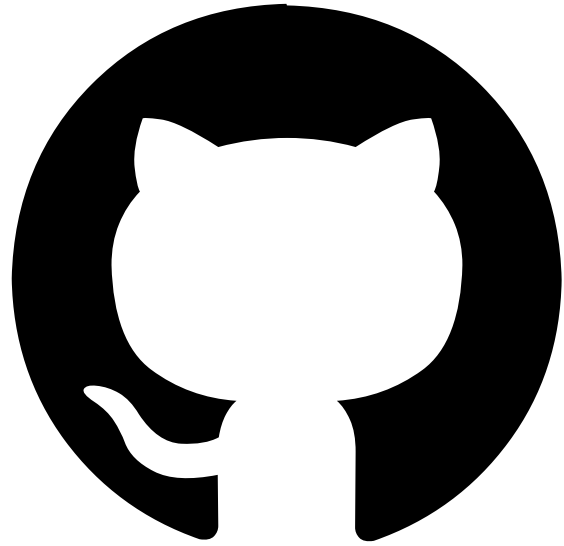
The [GitHub API documentation](#) has everything you need to integrate your tool or application with GitHub.



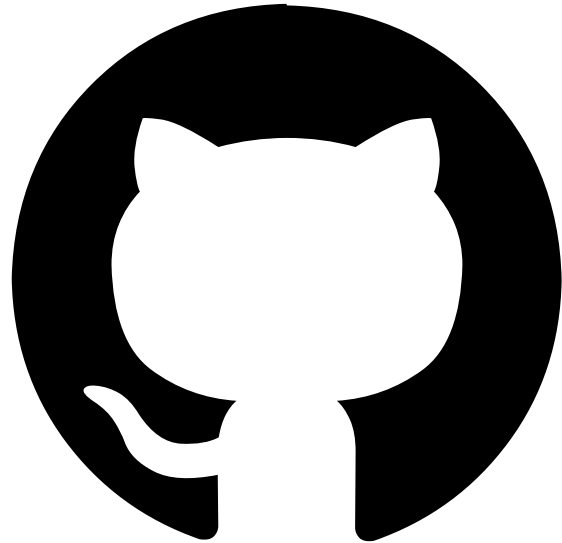


Powerful integrations





Any Questions?



Links

[Atlassian Git Guide](#)

[Git Immersion](#)

[GitHub Online Guides](#)

[Pro Git Book](#)