

WHY?

THE WALL STREET JOURNAL.

U.S. Edition ▾ | June 5, 2018 | Today's Paper | Video

Home World U.S. Politics Economy Business Tech Markets Opinion Life & Arts Real Estate WSJ. Magazine



TECH
Apple Unveils Ways to Help Limit iPhone Usage



TECH
Seven Things Apple Introduced That Could Improve Your Life



OPINION
Climate Change Has Run Its Course



MARKETS
When Markets Move, Here's Who Moves the Market



TECH
Microsoft GitHub in Stock

ESSAY

Why Software Is Eating The World

By Marc Andreessen

August 20, 2011

This week, Hewlett-Packard (where I am on the board) announced that it is exploring jettisoning its struggling PC business in favor of investing more heavily in software, where it sees better potential for growth. Meanwhile, Google plans to buy up the cellphone handset maker Motorola Mobility. Both moves surprised the tech world. But both moves are also in line with a trend I've observed, one that makes me optimistic about the future growth of the American and world economies, despite the recent turmoil in the stock market.

In...

RELATED VIDEO

Recommended Videos

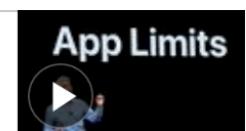
1. Apple iOS 12: The iPhone and iPad Updates You'll Care About



2. Deadly Volcano Erupts in Guatemala

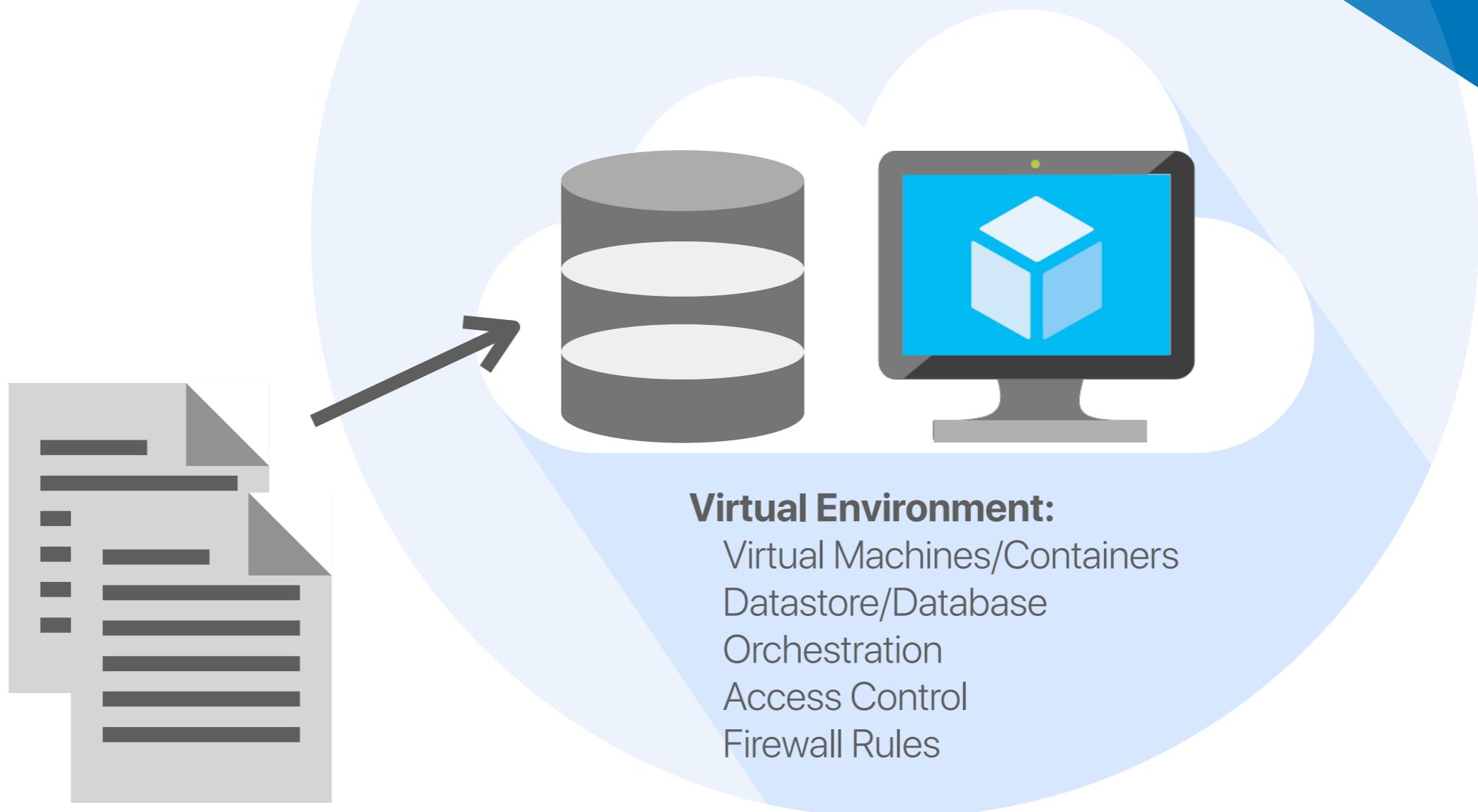


3. Apple iOS 12 Includes Features to Curb iPhone Addiction



**Clinical information systems
decisions will be software
decisions**

“Infrastructure as Code”





"Given how much we spend and how much technology we apply, I was struck by how much manual processing is going on. On the issue of measurement burden, that is an area for significant improvement. I wince when I see how much physician time is devoted to data entry. I think we can all agree that makes no sense in 2017 and we can do better."

—Lewis Sandy, M.D.
UnitedHealth, EVP for Clinical Advancement

“One of the reasons it takes so long for a drug to get to market, or any finding to lead to a change in clinical care is because of this lengthy cycle we go through of hoarding our data and hiding our algorithms in order to publish a paper so we can be first and get tenure and promotion. The patients deserve better than that. If we can address the attribution and contribution problem, you can be first just by posting it on GitHub or the web.”

—Melissa Haendel, Ph.D.
Department of Medical Informatics OHSU
National Center for Data to Health (CD2H)

"Every American needs to have the totality of the medical information available to every physician and hospital at every point of contact. It can be done. It's called ATMs. But it won't get done. Why is that? Because the people who manufacture and sell the electronic health records are not going to open up what's called APIs, the application processing software that's necessary for third-party developers to come in, because they know it will break the stranglehold they have on those who have purchased the systems already."

—Robert Pearl, M.D.
former CEO, The Permanente Medical Group

THE WALL STREET JOURNAL.

U.S. Edition ▾ | June 5, 2018 | Today's Paper | Video

Home World U.S. Politics Economy Business Tech Markets Opinion Life & Arts Real Estate WSJ. Magazine



TECH
Apple Unveils Ways to Help Limit iPhone Usage



TECH
Seven Things Apple Introduced That Could Improve Your Life



OPINION
Climate Change Has Run Its Course



MARKETS
When Markets Move, Here's Who Moves the Market



TECH
Microsoft GitHub in Stock

ESSAY

Why Software Is Eating The World

By Marc Andreessen

August 20, 2011

This week, Hewlett-Packard (where I am on the board) announced that it is exploring jettisoning its struggling PC business in favor of investing more heavily in software, where it sees better potential for growth. Meanwhile, Google plans to buy up the cellphone handset maker Motorola Mobility. Both moves surprised the tech world. But both moves are also in line with a trend I've observed, one that makes me optimistic about the future growth of the American and world economies, despite the recent turmoil in the stock market.

In...

RELATED VIDEO

Recommended Videos

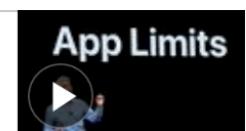
1. Apple iOS 12: The iPhone and iPad Updates You'll Care About



2. Deadly Volcano Erupts in Guatemala



3. Apple iOS 12 Includes Features to Curb iPhone Addiction







myFile_version01.txt



myFile_version02.txt



myFile_version03.txt



myFile_version05.txt



myFile_version06.txt



myFile_version01.txt



myFile_version02.txt



myFile_version03.txt



myFile_version05.txt



myFile_version06.txt



myFile_version01.txt



myFile_version02.txt



myFile_version03.txt



myFile_version05.txt



myFile_version06.txt



myFile_051518.txt



myFile_0515_2_18.txt



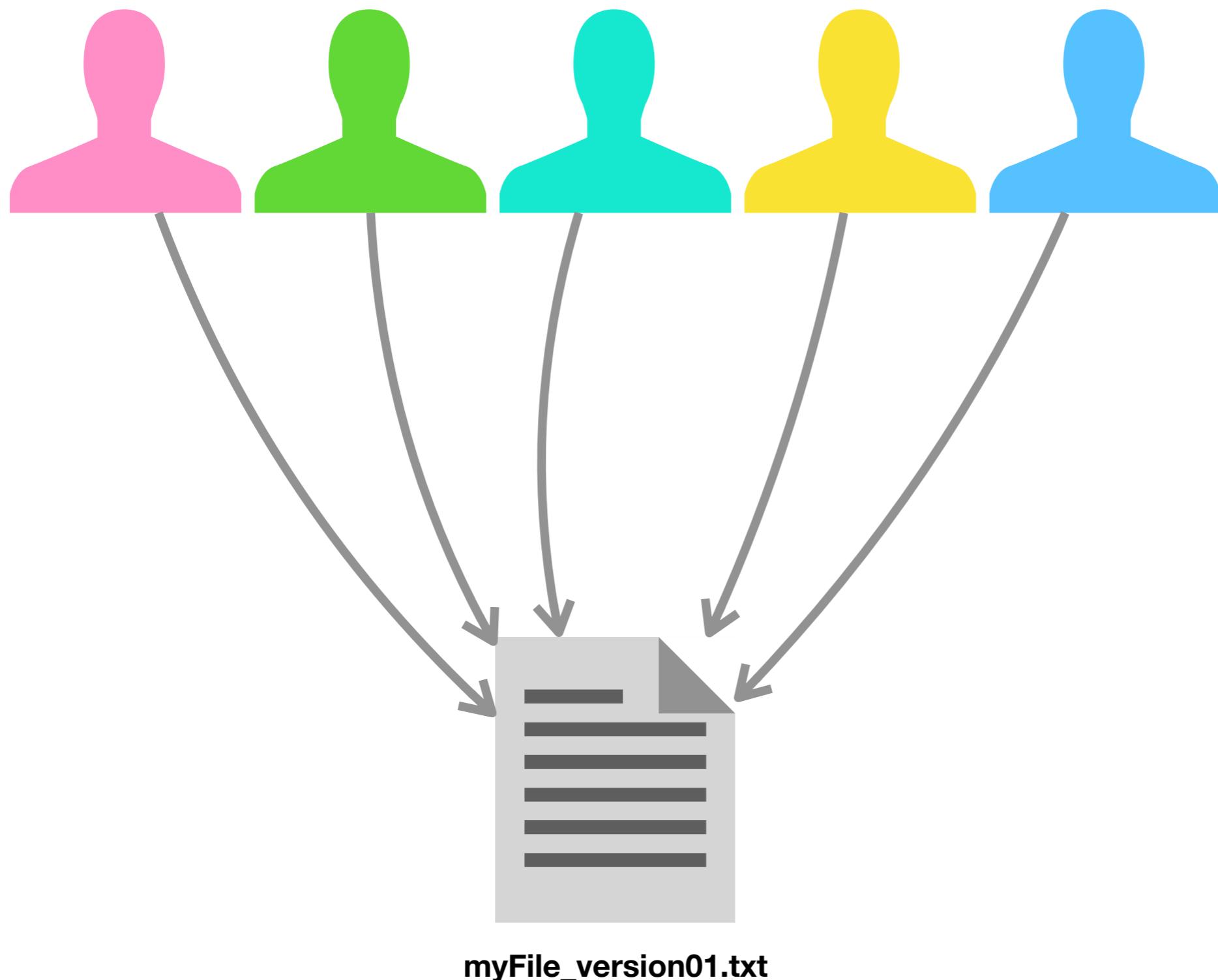
myFile_0515_3_18.txt

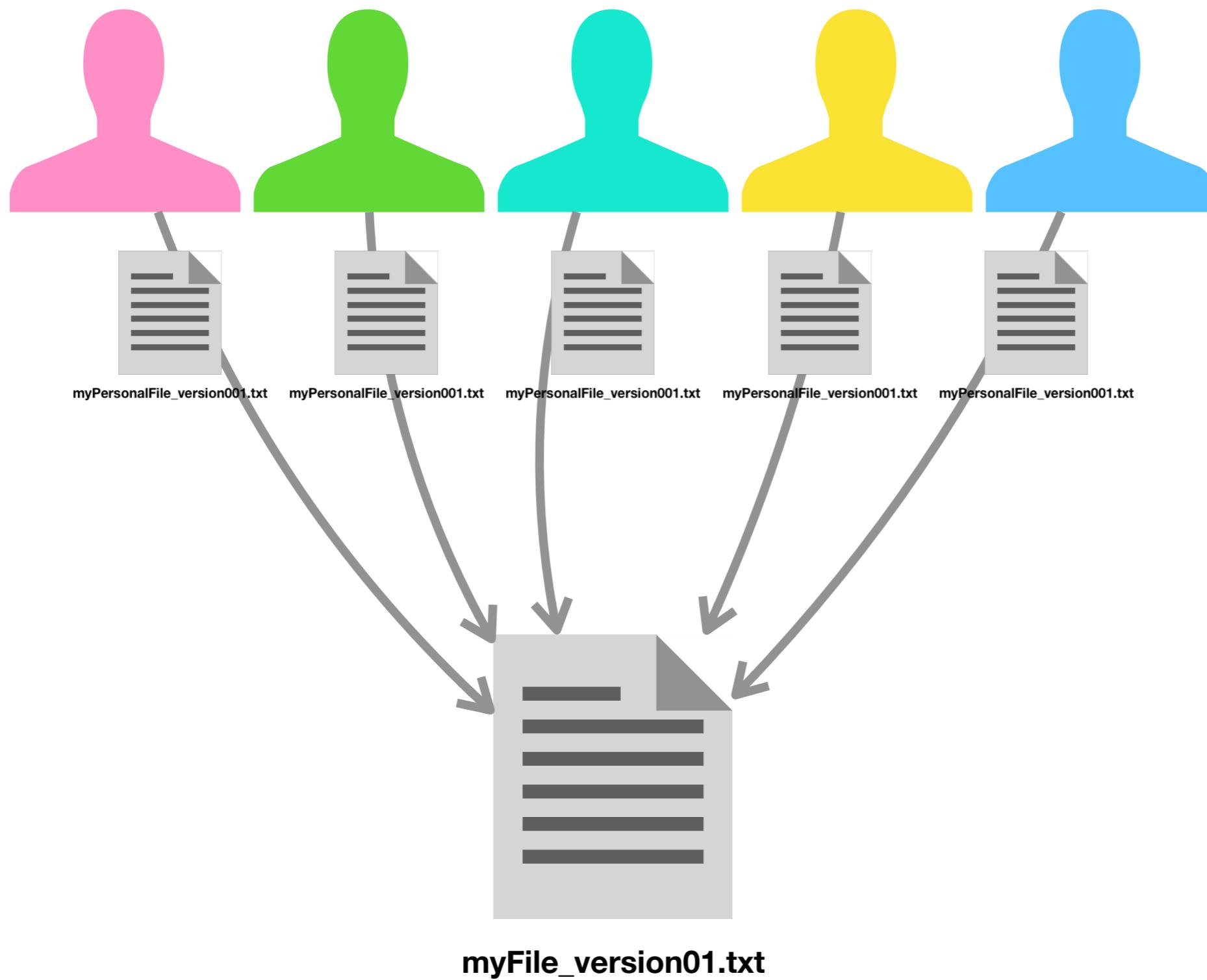


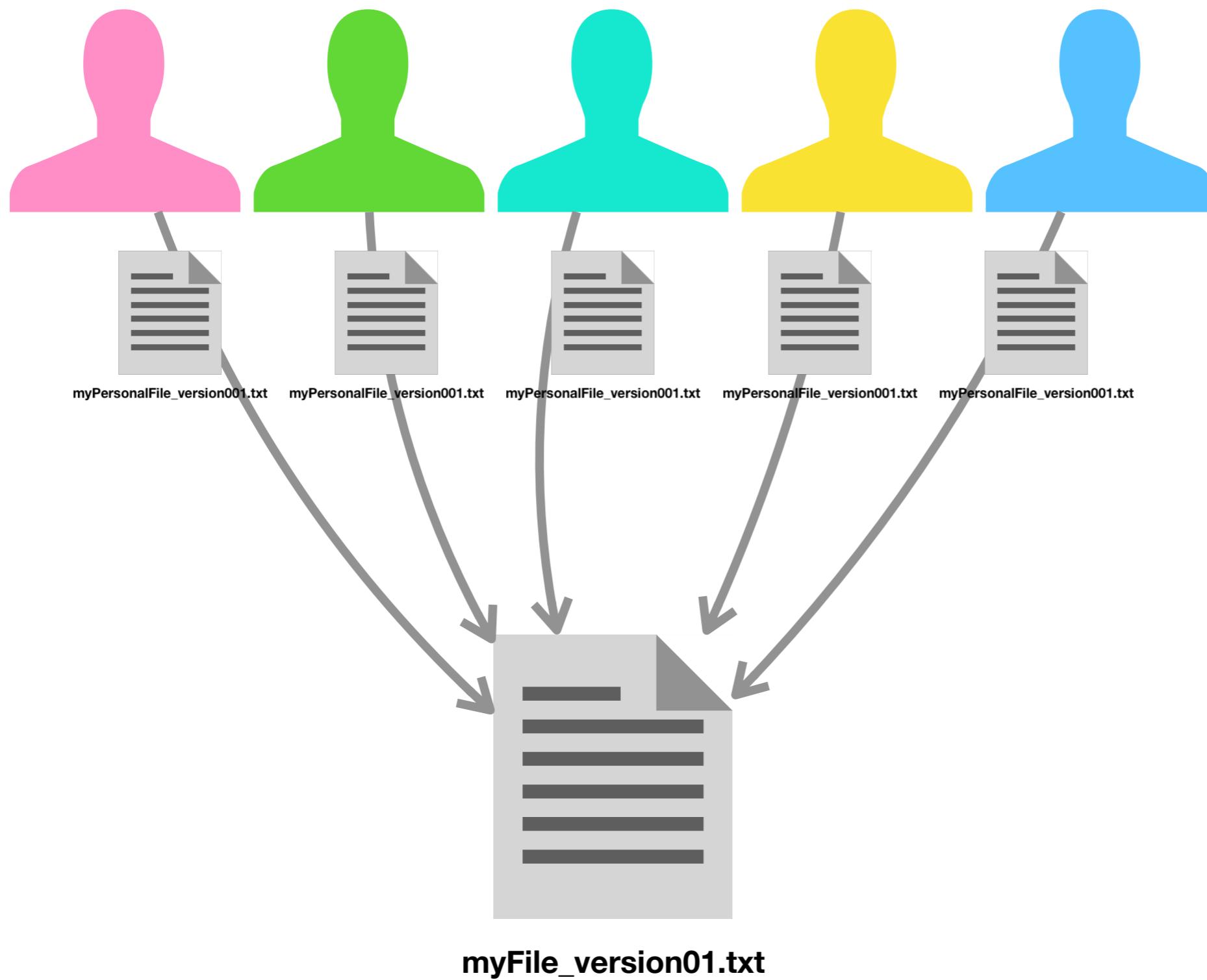
myFile_051618.txt



myFile_myLifeSucks.txt







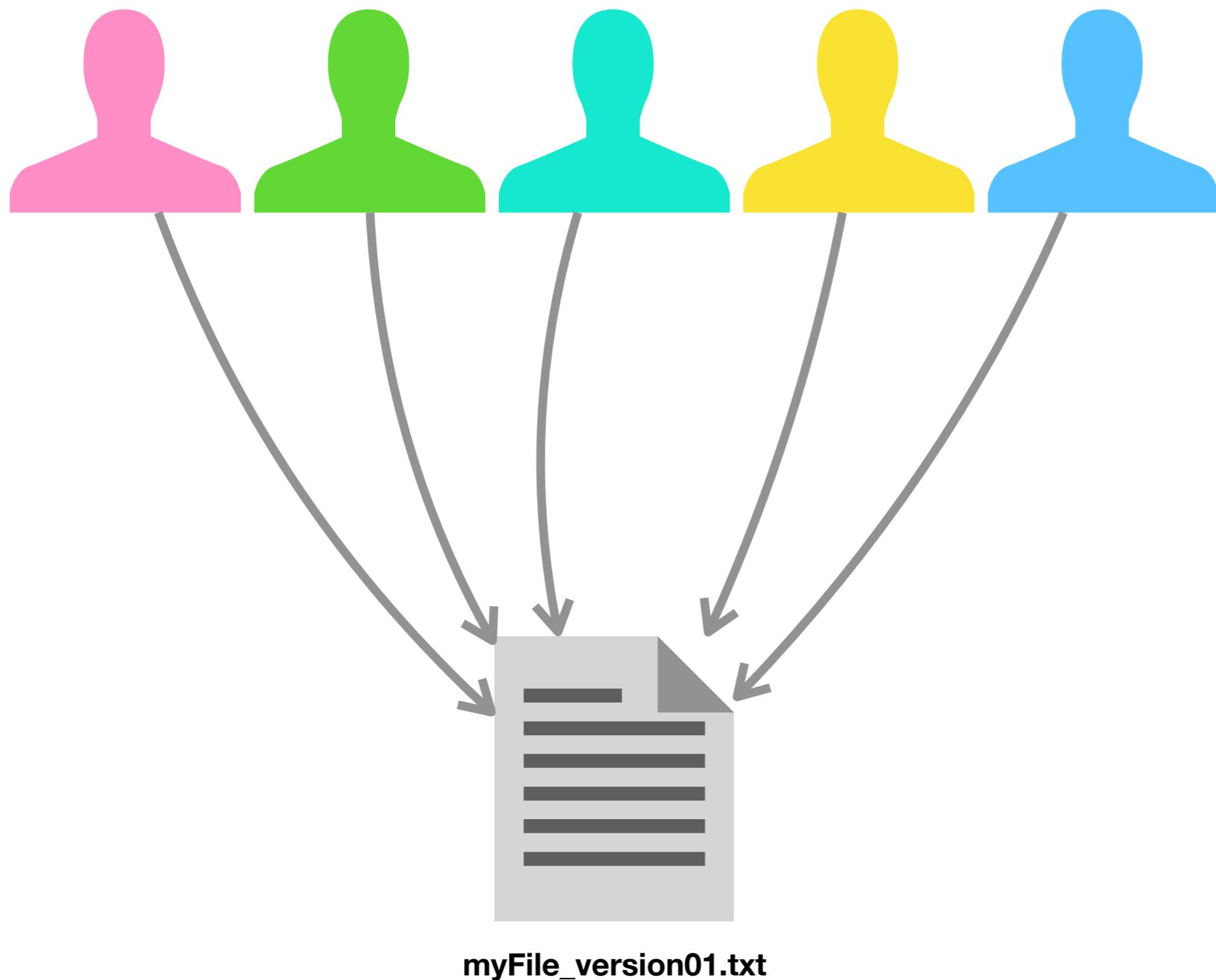
Why does Git work so well for Linux?

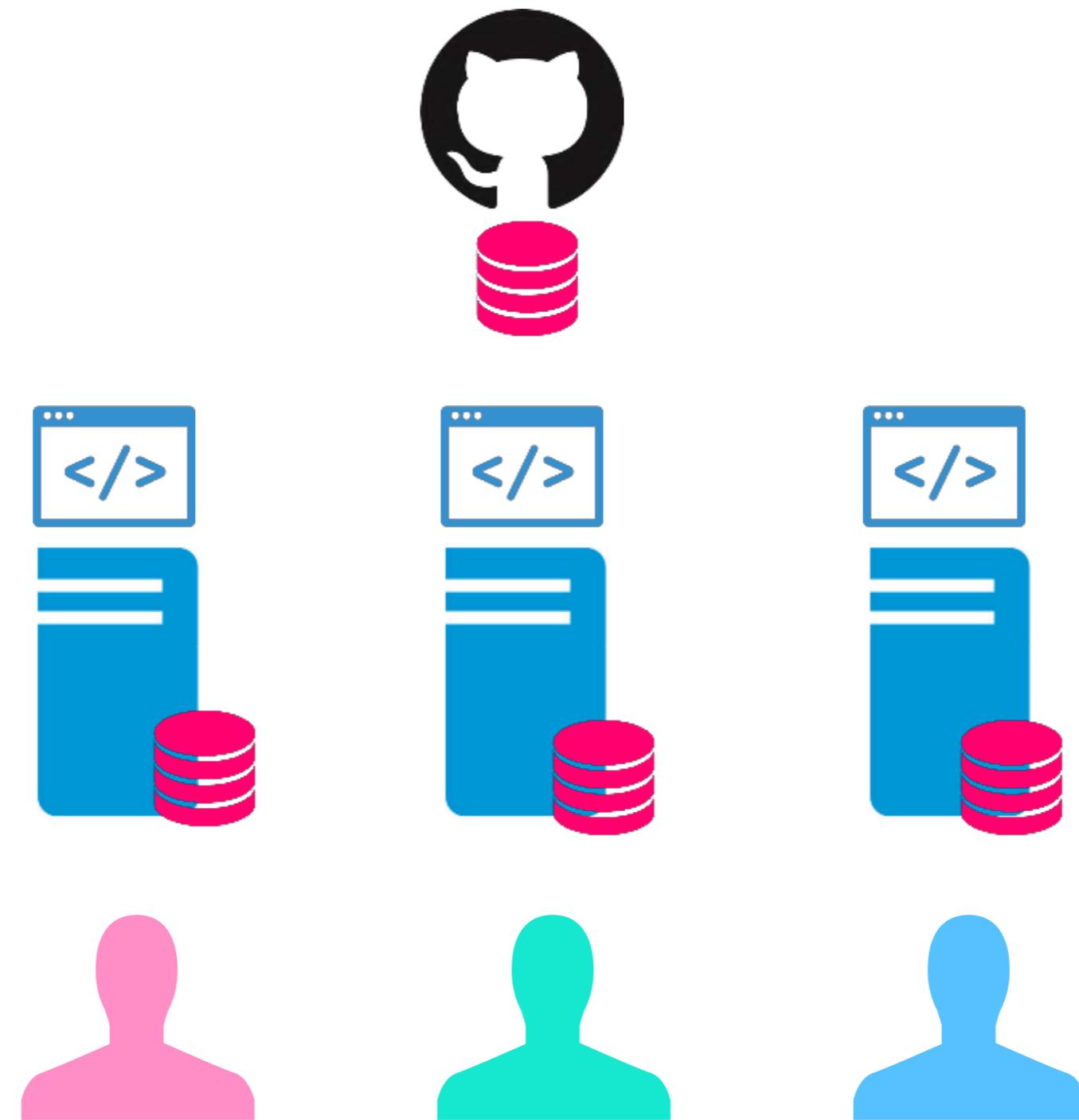
Torvalds: Well, it was obviously designed for our workflow, so that is part of it. I've already mentioned the whole "distributed" part many times, but it bears repeating. But it was also designed to be efficient enough for a biggish project like Linux, and it was designed to do things that people considered "hard" before git – because those are the things I do every day.

Just to pick an example: **the concept of "merging"** was generally considered to be something really quite painful and hard in most SCM's. You'd plan your merges, because they were big deals. That's not acceptable to me, since I commonly do tens of merges a day when in the merge window, and even then, the biggest overhead shouldn't be the merge itself, it should be testing the result. The "git" part of the merge is just a couple of seconds, **it should take me much longer just to write the merge explanation message.**

So git was basically designed and written for my requirements, and it shows.

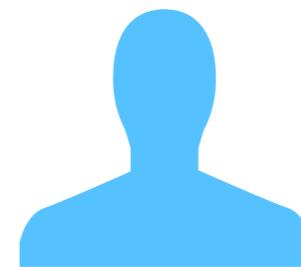
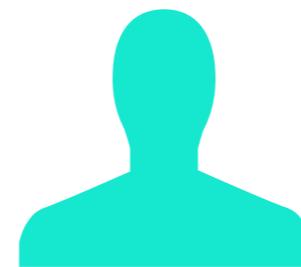
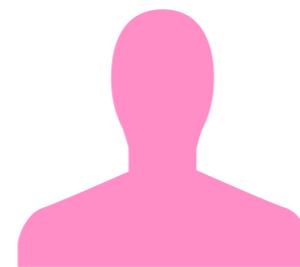
<https://github.com/torvalds/linux>

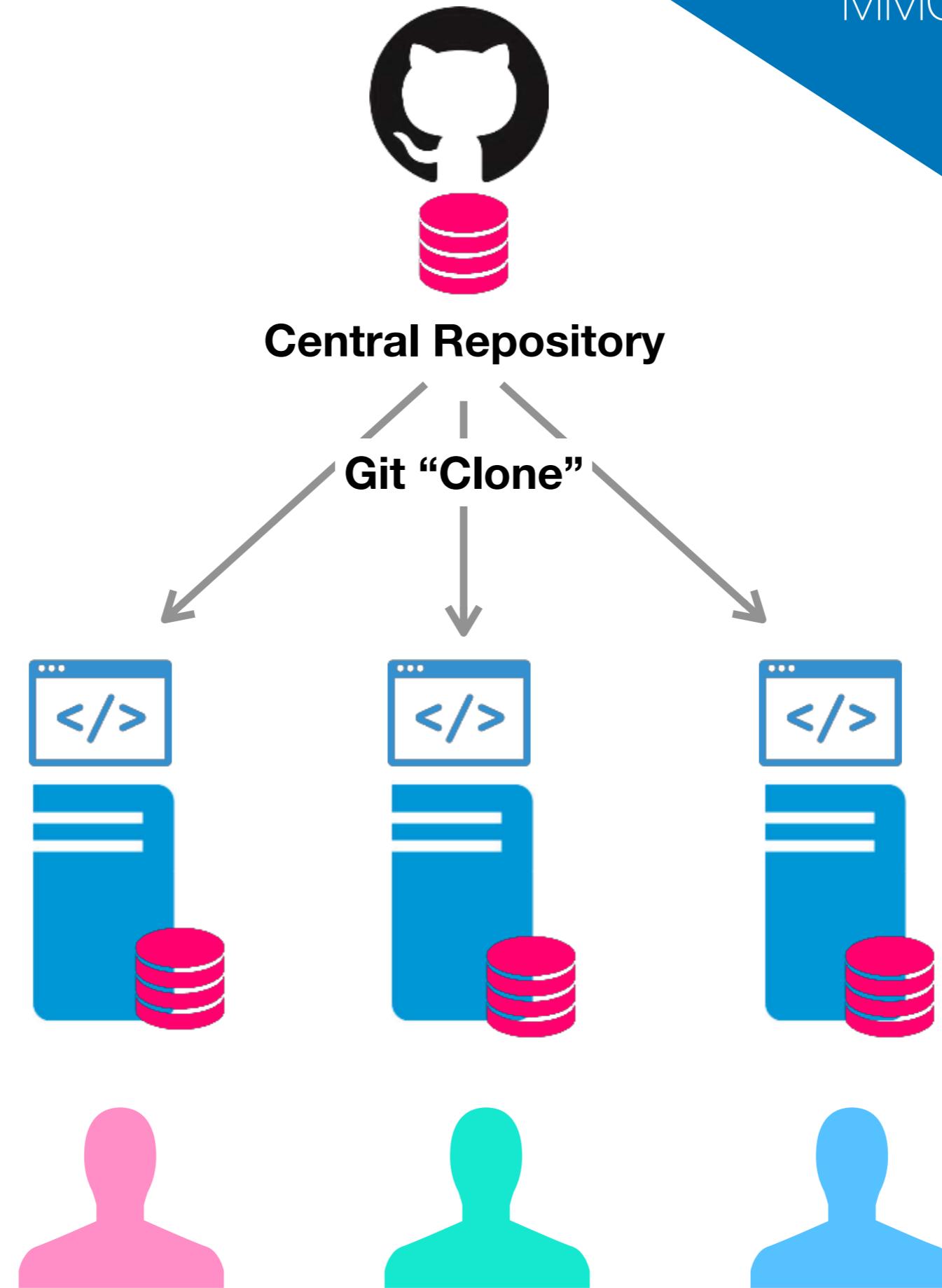




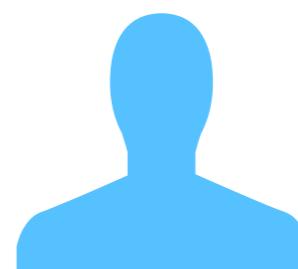
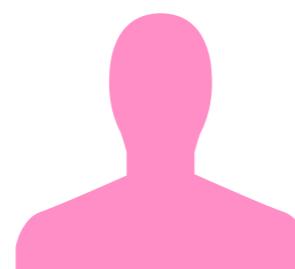
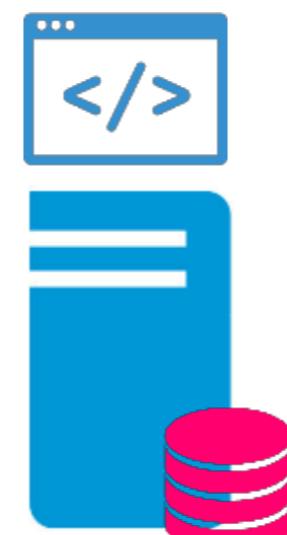
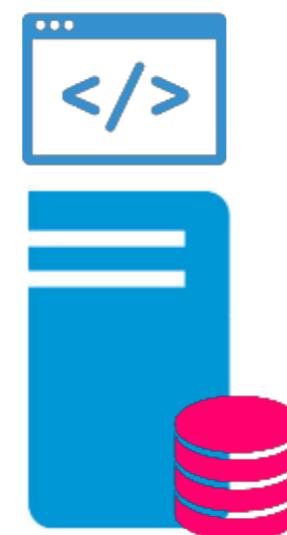


Central Repository

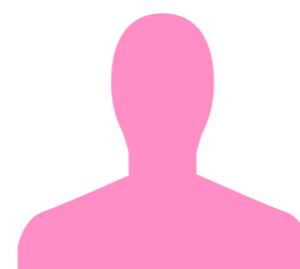
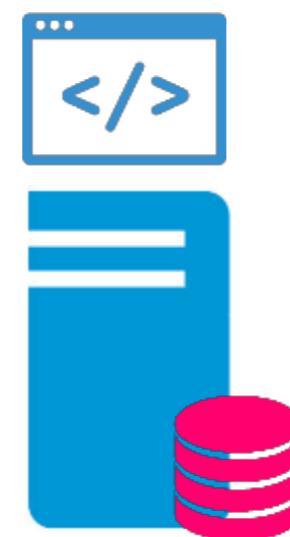


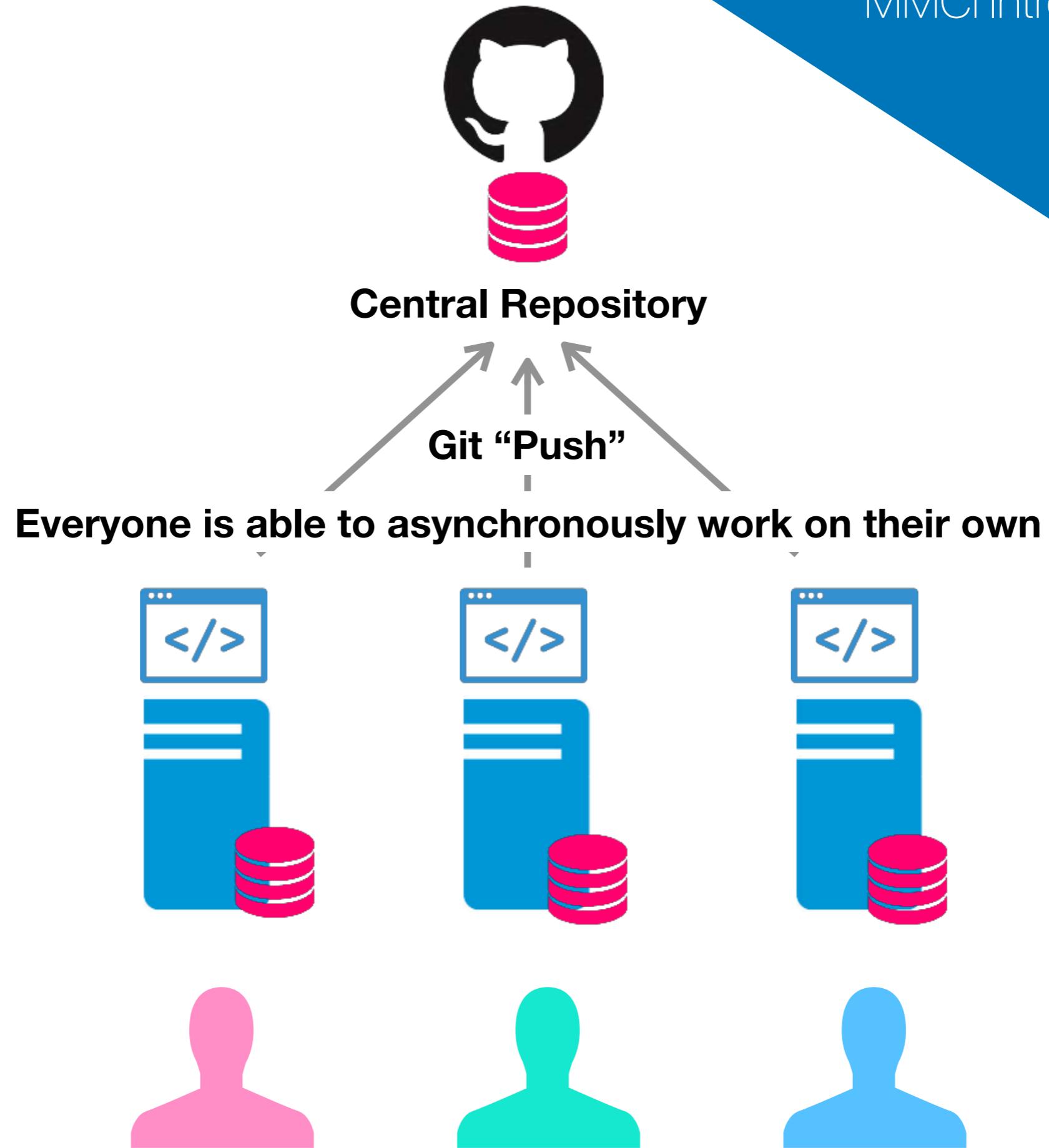


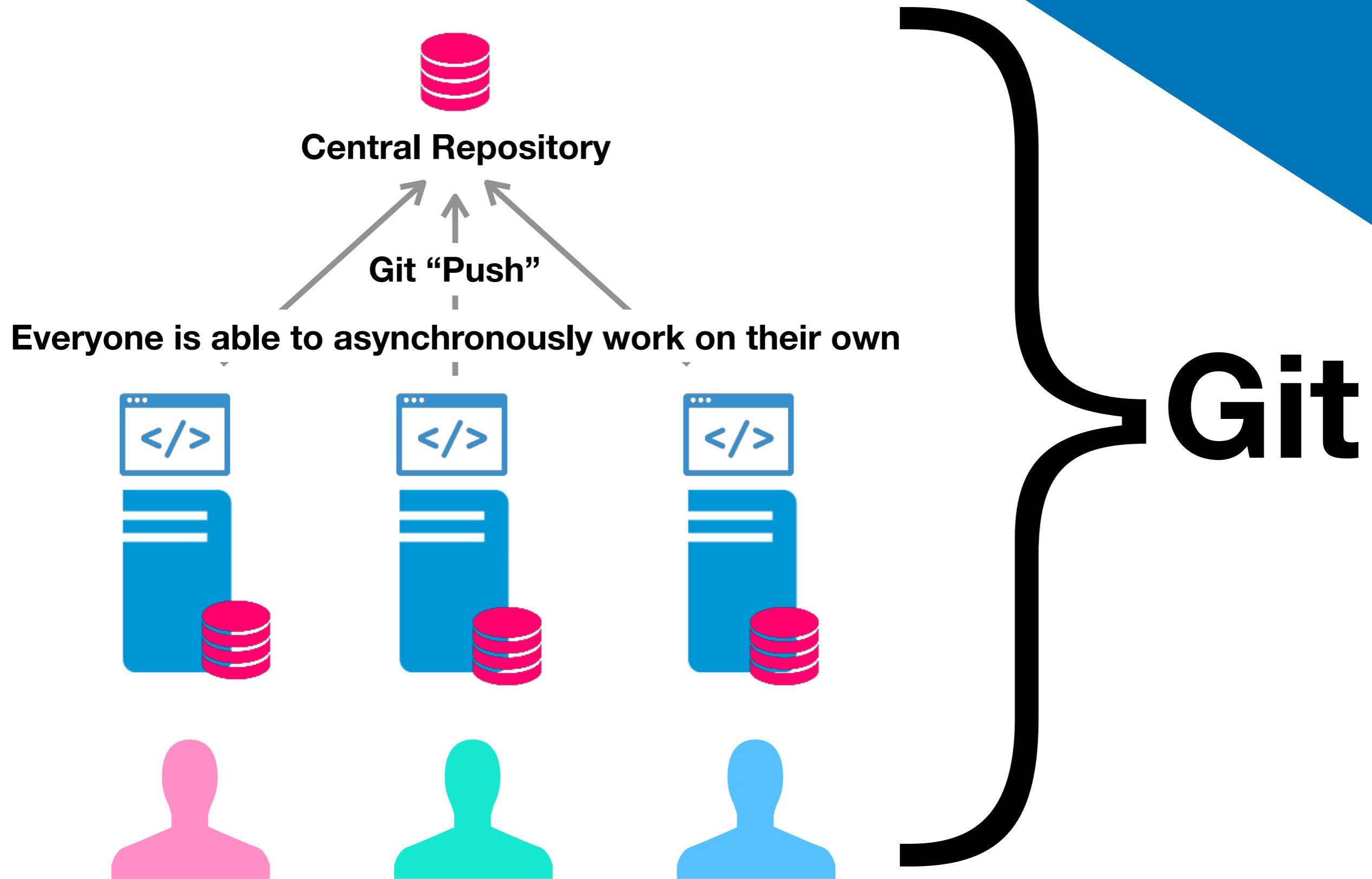
Everyone is able to asynchronously work on their own

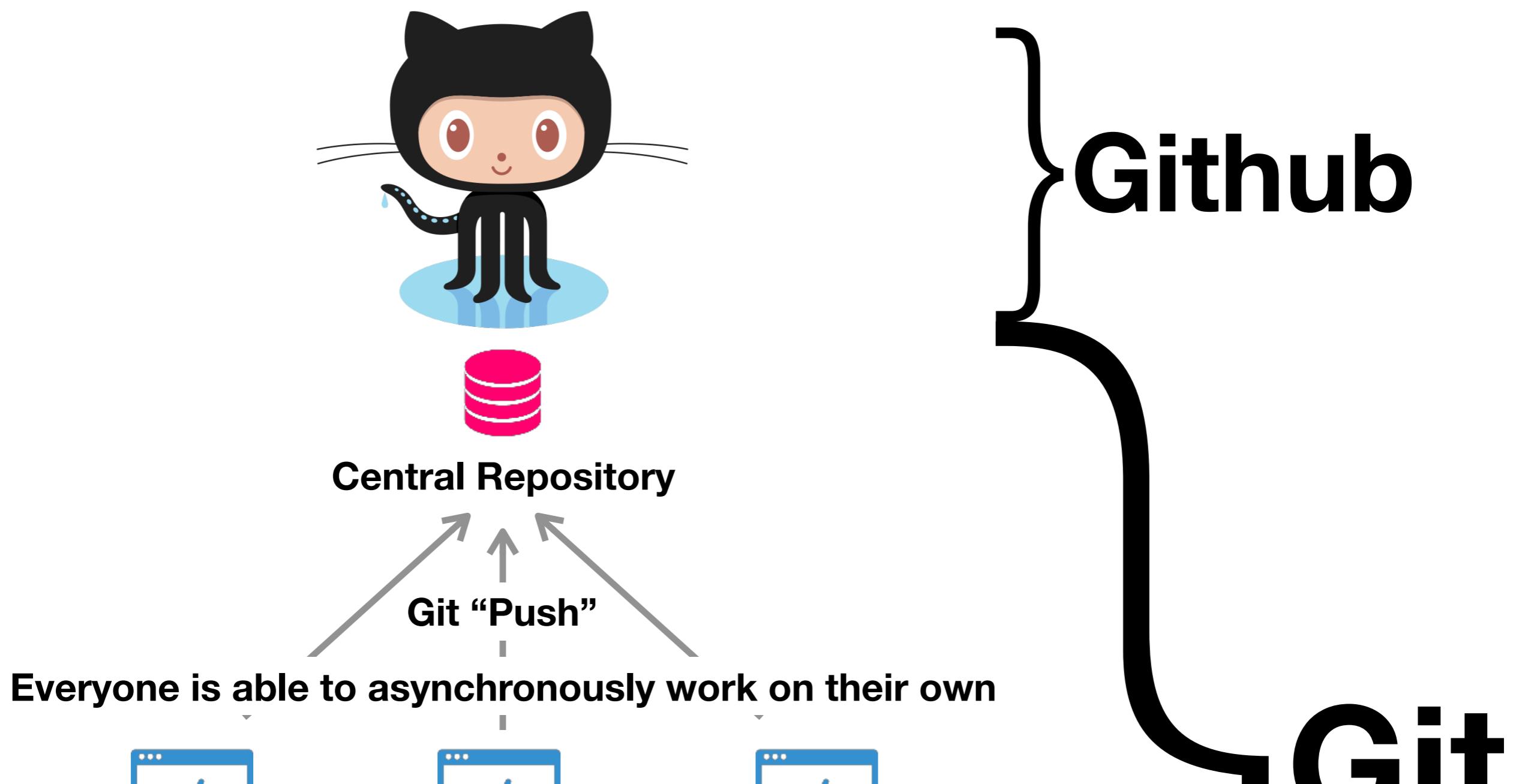


Everyone is able to asynchronously work on their own











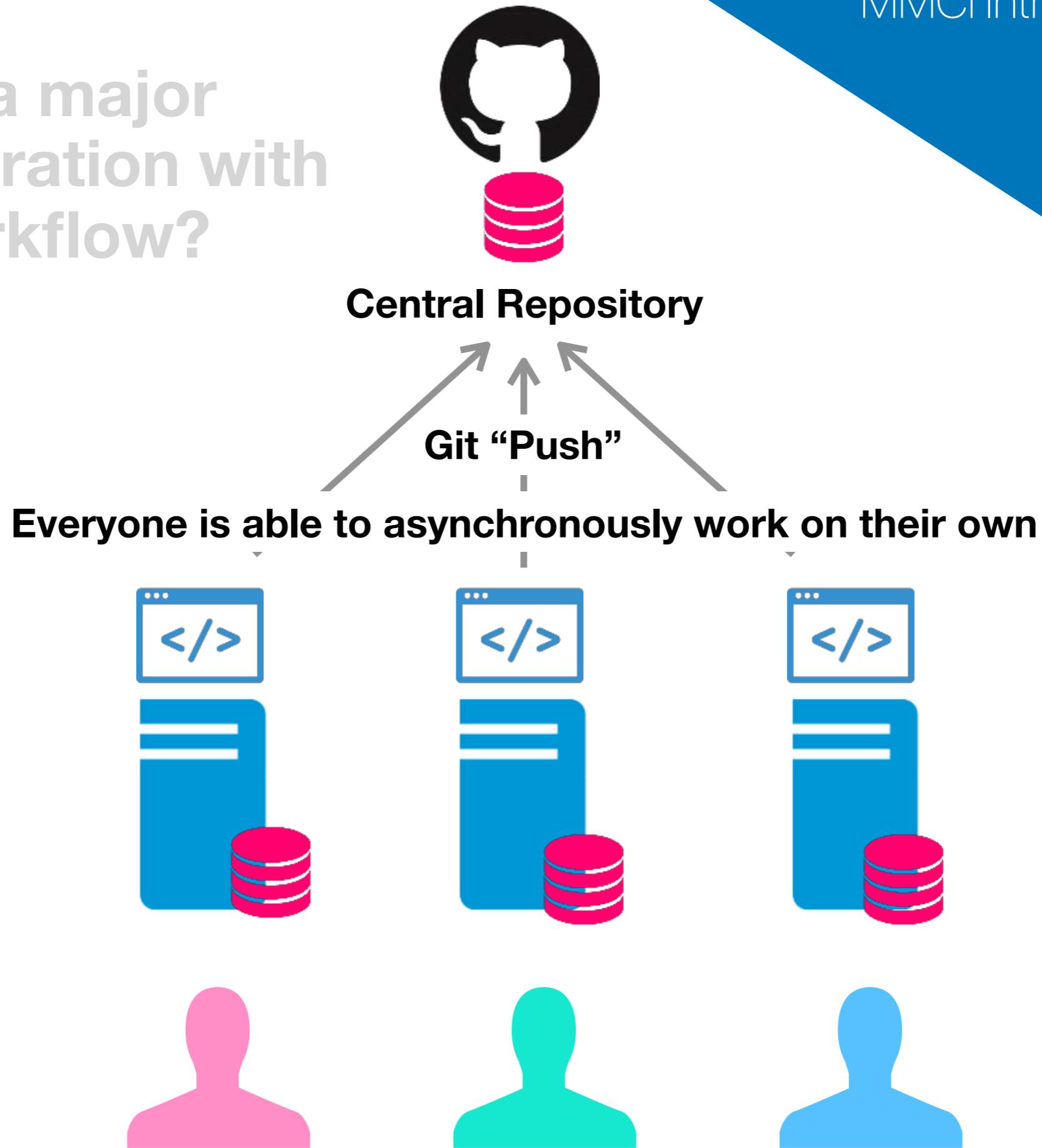
GitLab



Web Front Ends & Tooling for Git

Why Github/lab?

What's a major consideration with this workflow?



Simple workflow for getting started

The screenshot shows the GitHub organization page for "Duke Translational Bioinformatics". The top navigation bar includes the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. The organization's logo, a blue circle with a white "TX" and arrows, is displayed. The main content area lists three repositories: "tesseract", "duke-data-service-portal", and "duke-data-service". Each repository card includes the name, description, language, stars, forks, and last update. To the right, there are sections for "Top languages" (R, Python, JavaScript, Jupyter Notebook, Ruby) and "People" (27). A red arrow points to the "New" button in the pinned repositories section.

Duke Translational Bioinformatics

Repositories 90 People 27 Teams 13 Projects 1 Settings

Search repositories... Type: All Language: All Customize pinned repositories New

tesseract
Forked from tesseract-ocr/tesseract
Tesseract Open Source OCR Engine (main repository)
C++ 3,868 Updated 3 days ago

duke-data-service-portal
A web portal built on the Duke Data Service API
JavaScript 7 GPL-3.0 3 issues need help Updated 8 days ago

duke-data-service
Main repository for the Duke Data Service
Ruby 10 13 GPL-3.0 Updated 8 days ago

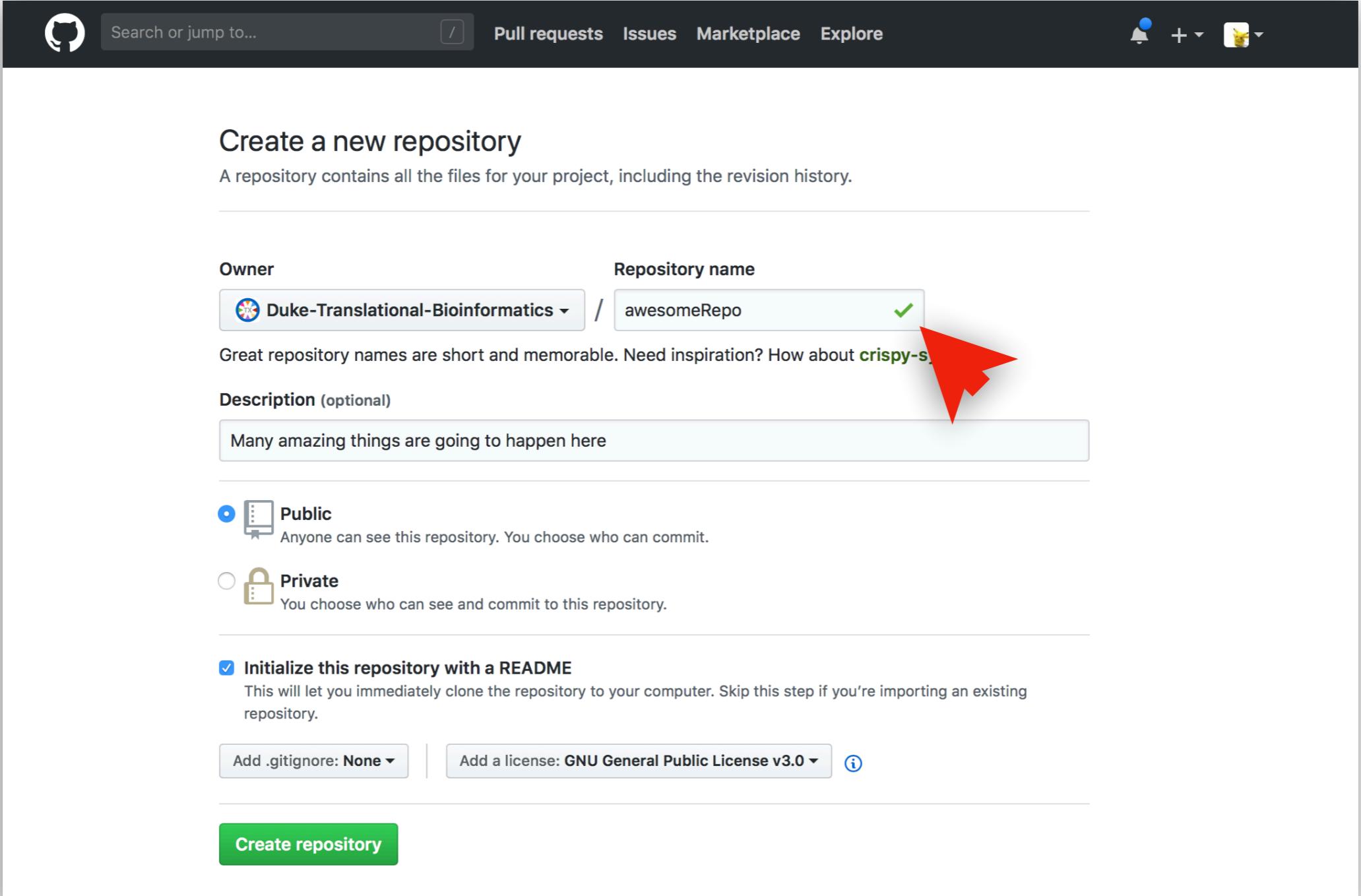
Top languages

- R
- Python
- JavaScript
- Jupyter Notebook
- Ruby

People 27

Invite someone

Simple workflow for getting started



The screenshot shows the GitHub interface for creating a new repository. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. On the right side of the bar are icons for notifications, a plus sign, and a user profile.

The main section is titled "Create a new repository". It explains that a repository contains all the files for a project, including the revision history. Below this, there are fields for "Owner" (set to "Duke-Translational-Bioinformatics") and "Repository name" (set to "awesomeRepo"). A red arrow points to the "Repository name" field. A suggestion "Great repository names are short and memorable. Need inspiration? How about crispy-s..." is visible next to the input field.

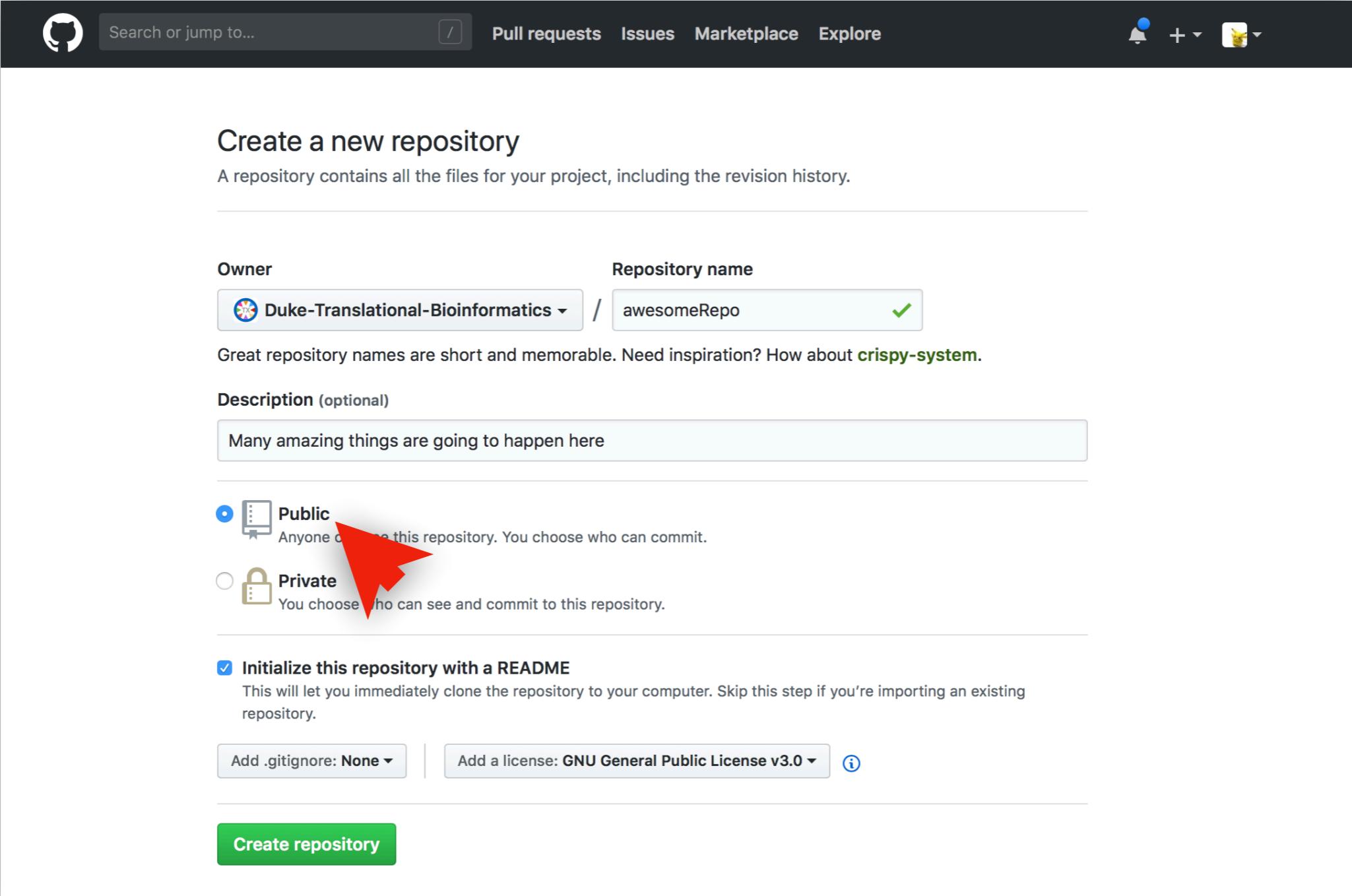
The "Description (optional)" field contains the text "Many amazing things are going to happen here".

Below these fields are two radio button options for visibility: "Public" (selected) and "Private". The "Public" option is described as allowing anyone to see the repository while letting the user choose who can commit. The "Private" option is described as letting the user choose who can see and commit to the repository.

A checked checkbox labeled "Initialize this repository with a README" is present. A explanatory text below it states: "This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository." There are also dropdown menus for "Add .gitignore: None" and "Add a license: GNU General Public License v3.0".

At the bottom, a large green button with the text "Create repository" is centered.

Simple workflow for getting started



The screenshot shows the GitHub interface for creating a new repository. At the top, there's a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. On the right side of the bar are icons for notifications, a plus sign, and a user profile.

The main section is titled "Create a new repository". It explains that a repository contains all the files for your project, including the revision history. Below this, there are fields for "Owner" (set to "Duke-Translational-Bioinformatics") and "Repository name" (set to "awesomeRepo"). A green checkmark icon is next to the repository name field.

A note below the name field suggests using short and memorable names, with "crispy-system" as an example.

The "Description (optional)" field contains the placeholder text "Many amazing things are going to happen here".

The "Visibility" section shows two options: "Public" (selected, indicated by a blue dot) and "Private" (indicated by a red arrow pointing to it). The "Public" option is described as allowing anyone to clone the repository and choose who can commit. The "Private" option is described as allowing the user to choose who can see and commit to the repository.

The "Initialize this repository with a README" checkbox is checked, with a note explaining it lets you immediately clone the repository to your computer. There are dropdown menus for ".gitignore" (set to "None") and "Add a license" (set to "GNU General Public License v3.0").

At the bottom is a large green "Create repository" button.

Simple workflow for getting started

The screenshot shows the GitHub interface for creating a new repository. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. On the right side of the bar are icons for notifications, a plus sign, and a user profile.

The main section is titled "Create a new repository". It explains that a repository contains all the files for your project, including the revision history. Below this, there are fields for "Owner" (set to "Duke-Translational-Bioinformatics") and "Repository name" (set to "awesomeRepo"). A green checkmark icon is next to the repository name field.

A note below the name field suggests using short and memorable names, mentioning "crispy-system" as an example.

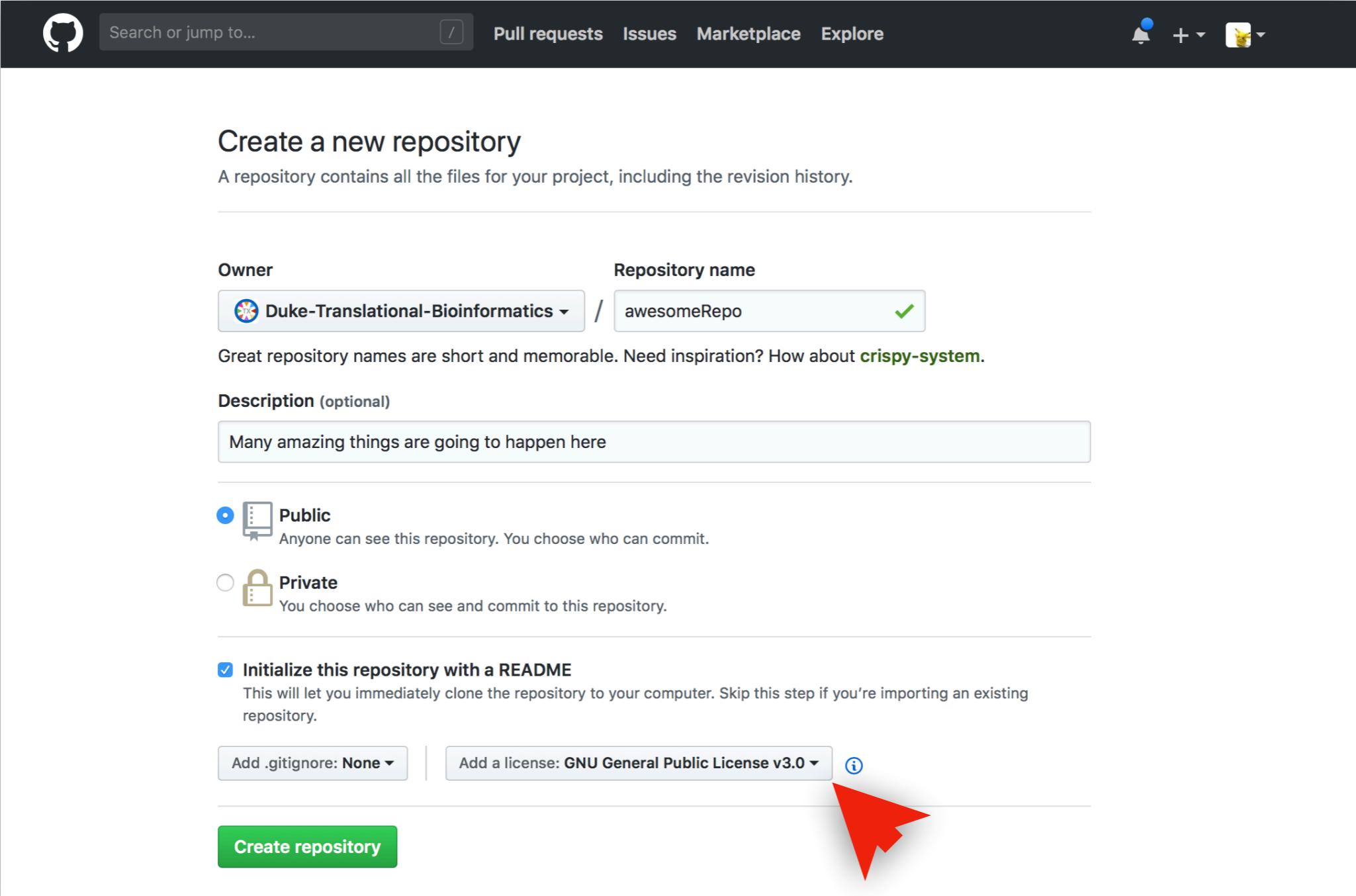
The "Description (optional)" field contains the placeholder text "Many amazing things are going to happen here".

Below the description, there are two radio button options for repository visibility: "Public" (selected) and "Private". The "Public" option is described as allowing anyone to see the repository while letting the user choose who can commit. The "Private" option is described as letting the user choose who can see and commit to the repository.

At the bottom of the form, there is a checked checkbox labeled "Initialize this repository with a README". A red arrow points to this checkbox. A explanatory text below it says: "This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository." There are also dropdown menus for "Add .gitignore: None" and "Add a license: GNU General Public License v3.0".

At the very bottom is a large green "Create repository" button.

Simple workflow for getting started



The screenshot shows the GitHub interface for creating a new repository. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. On the right side of the bar are icons for notifications, a plus sign, and a user profile.

The main section is titled "Create a new repository". It explains that a repository contains all the files for a project, including the revision history. Below this, there are fields for "Owner" (set to "Duke-Translational-Bioinformatics") and "Repository name" (set to "awesomeRepo"). A green checkmark icon is next to the repository name field.

A note below the name field suggests using short and memorable names, mentioning "crispy-system" as an example.

The "Description (optional)" field contains the text "Many amazing things are going to happen here".

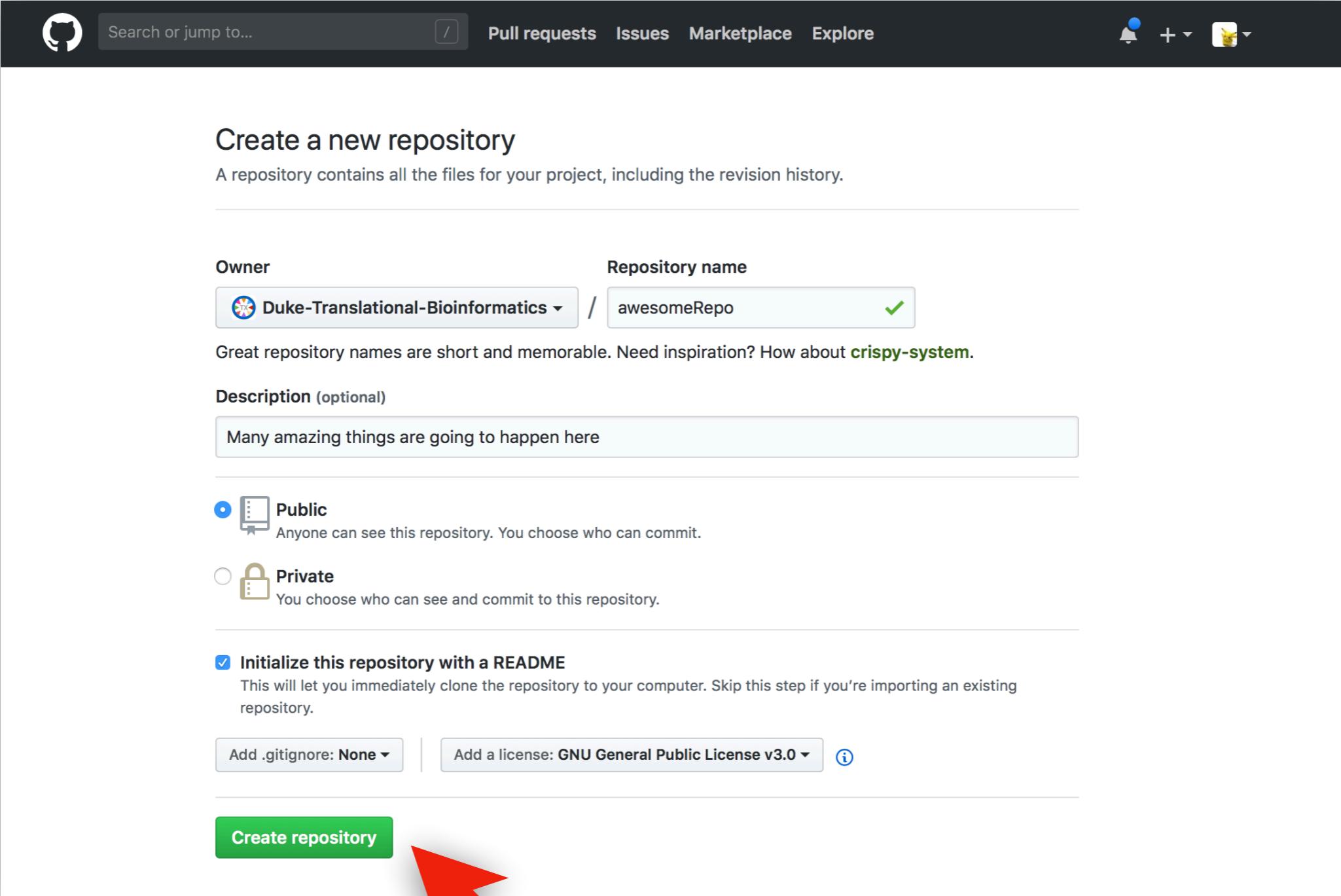
Below the description, there are two radio button options for repository visibility: "Public" (selected) and "Private". The "Public" option is described as allowing anyone to see the repository while letting the owner choose who can commit. The "Private" option is described as letting the owner choose who can see and commit to the repository.

At the bottom of the form, there is a checked checkbox for "Initialize this repository with a README". A explanatory text below it states that this will let you immediately clone the repository to your computer, skipping this step if you're importing an existing repository.

At the very bottom, there are two dropdown menus: "Add .gitignore: None" and "Add a license: GNU General Public License v3.0". To the right of the "Add a license" dropdown is a blue information icon with a white question mark.

The final step is the "Create repository" button, which is highlighted with a large red arrow pointing towards it from the bottom right.

Simple workflow for getting started



The screenshot shows the GitHub interface for creating a new repository. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. On the right side of the bar are icons for notifications, a plus sign, and a user profile.

The main section is titled "Create a new repository". It explains that a repository contains all the files for a project, including the revision history. Below this, there are fields for "Owner" (set to "Duke-Translational-Bioinformatics") and "Repository name" (set to "awesomeRepo"). A green checkmark icon is next to the repository name field.

A note below the name field suggests using short and memorable names, mentioning "crispy-system" as an example.

The "Description (optional)" field contains the text "Many amazing things are going to happen here".

Below the description, there are two radio button options for repository visibility: "Public" (selected) and "Private". The "Public" option is described as allowing anyone to see the repository while letting the owner choose who can commit. The "Private" option is described as letting the owner choose who can see and commit to the repository.

At the bottom of the form, there is a checked checkbox for "Initialize this repository with a README". A note explains that this will let you immediately clone the repository to your computer, skipping this step if you're importing an existing repository. There are also dropdown menus for "Add .gitignore: None" and "Add a license: GNU General Public License v3.0".

The bottom right corner of the screenshot features a large red arrow pointing towards the "Create repository" button, which is highlighted with a green background and white text.

Simple workflow for getting started

The screenshot shows a GitHub repository page for 'Duke-Translational-Bioinformatics / awesomeRepo'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. The latest commit was made by erichhuang just now. The repository contains files LICENSE and README.md, both of which are initial commits made just now. The README.md file contains the text 'Many amazing things are going to happen here'.

Search or jump to... / Pull requests Issues Marketplace Explore

Duke-Translational-Bioinformatics / **awesomeRepo** Unwatch 3 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Many amazing things are going to happen here Edit

Add topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

erichhuang Initial commit Latest commit 7e55bee just now

LICENSE Initial commit just now

README.md Initial commit just now

README.md

awesomeRepo

Many amazing things are going to happen here

Simple workflow for getting started

The screenshot shows a GitHub repository page for 'Duke-Translational-Bioinformatics / awesomeRepo'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. The latest commit was made by erichhuang just now. The repository contains files: LICENSE and README.md. A red arrow points to the 'Clone or download' button.

Search or jump to... / Pull requests Issues Marketplace Explore

Duke-Translational-Bioinformatics / **awesomeRepo** Unwatch 3 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Many amazing things are going to happen here Edit

Add topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

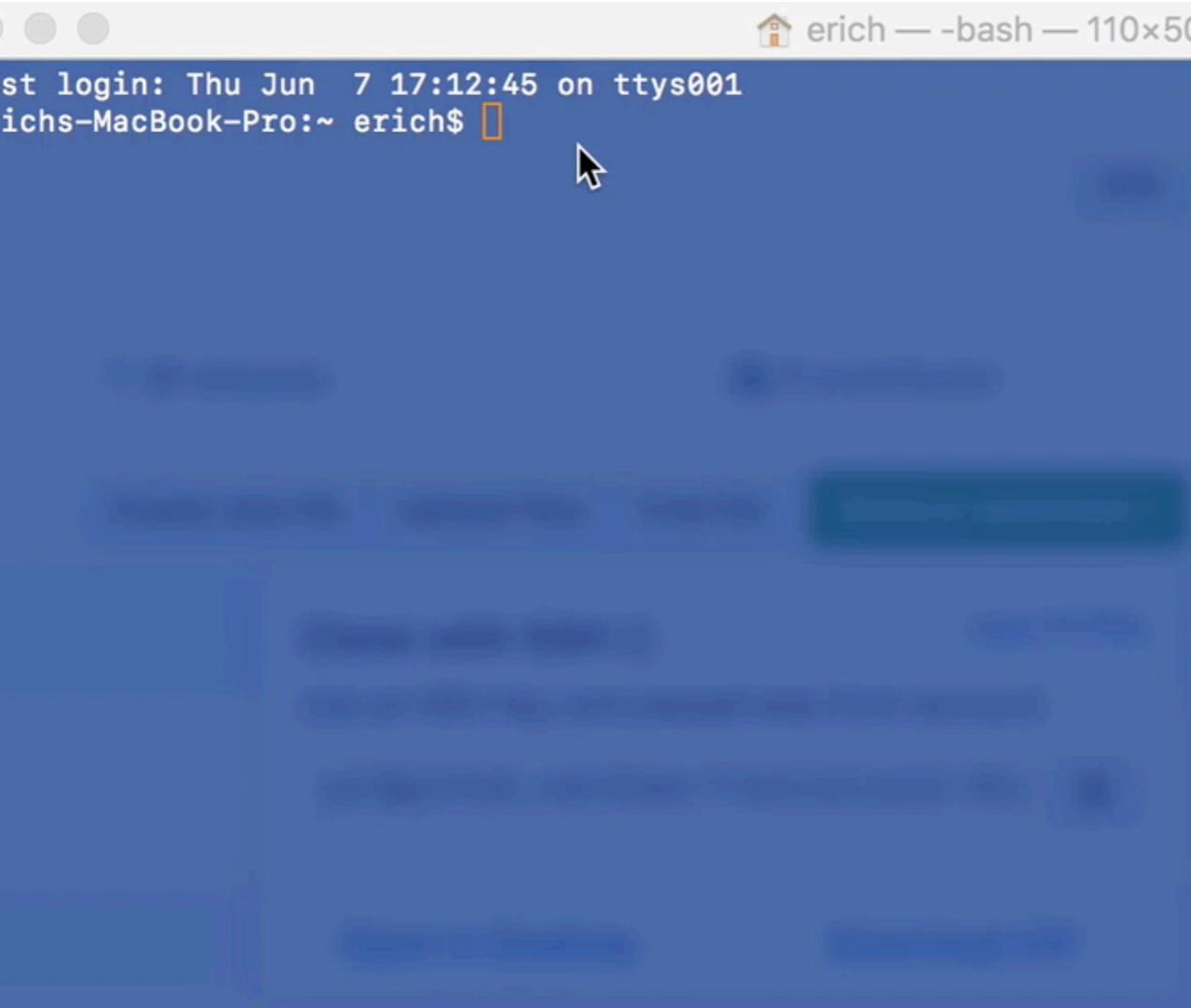
erichhuang Initial commit	Latest commit 7e55bee just now
LICENSE	Initial commit just now
README.md	Initial commit just now

README.md

awesomeRepo

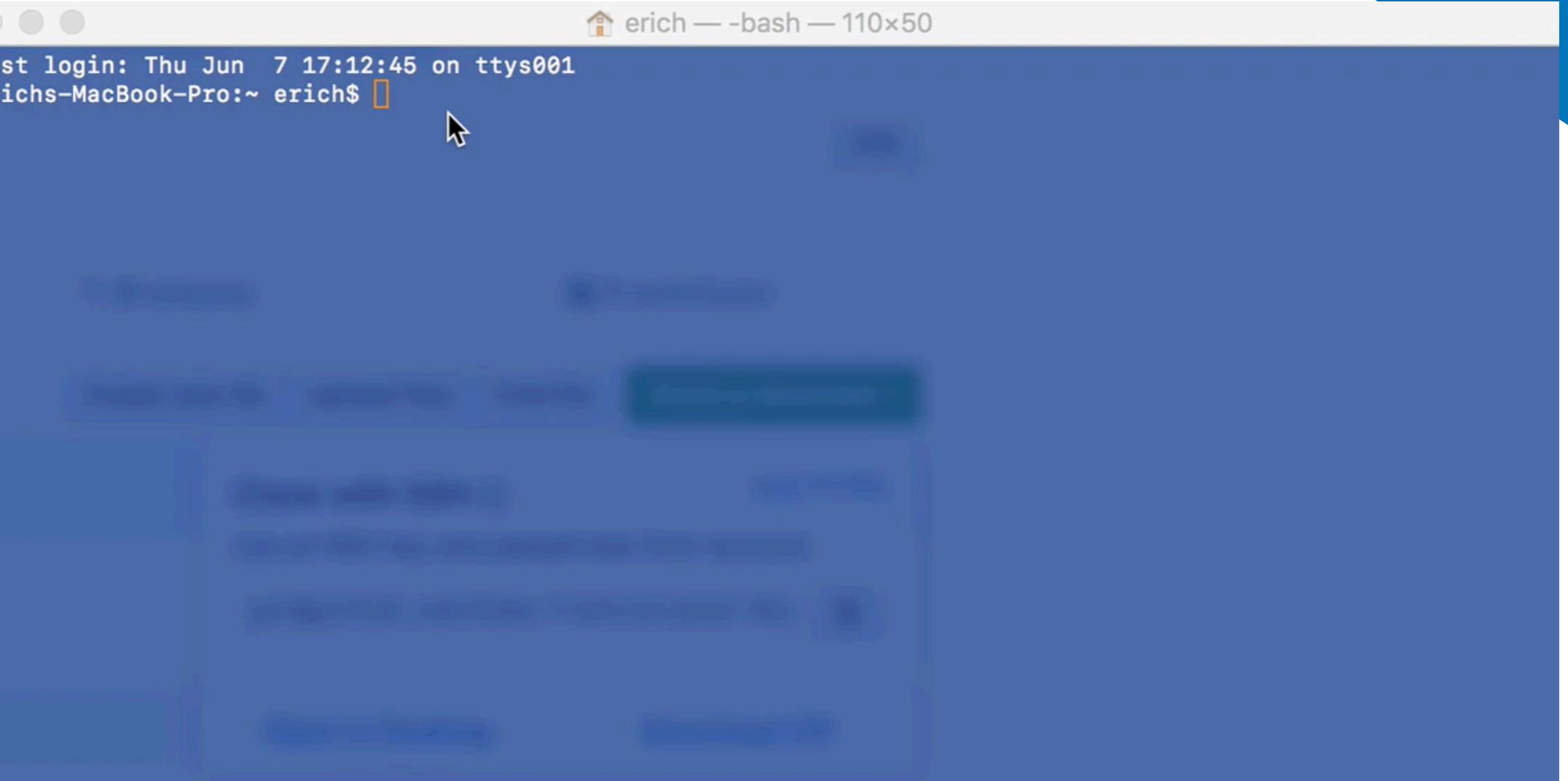
Many amazing things are going to happen here

Simple workflow for getting started



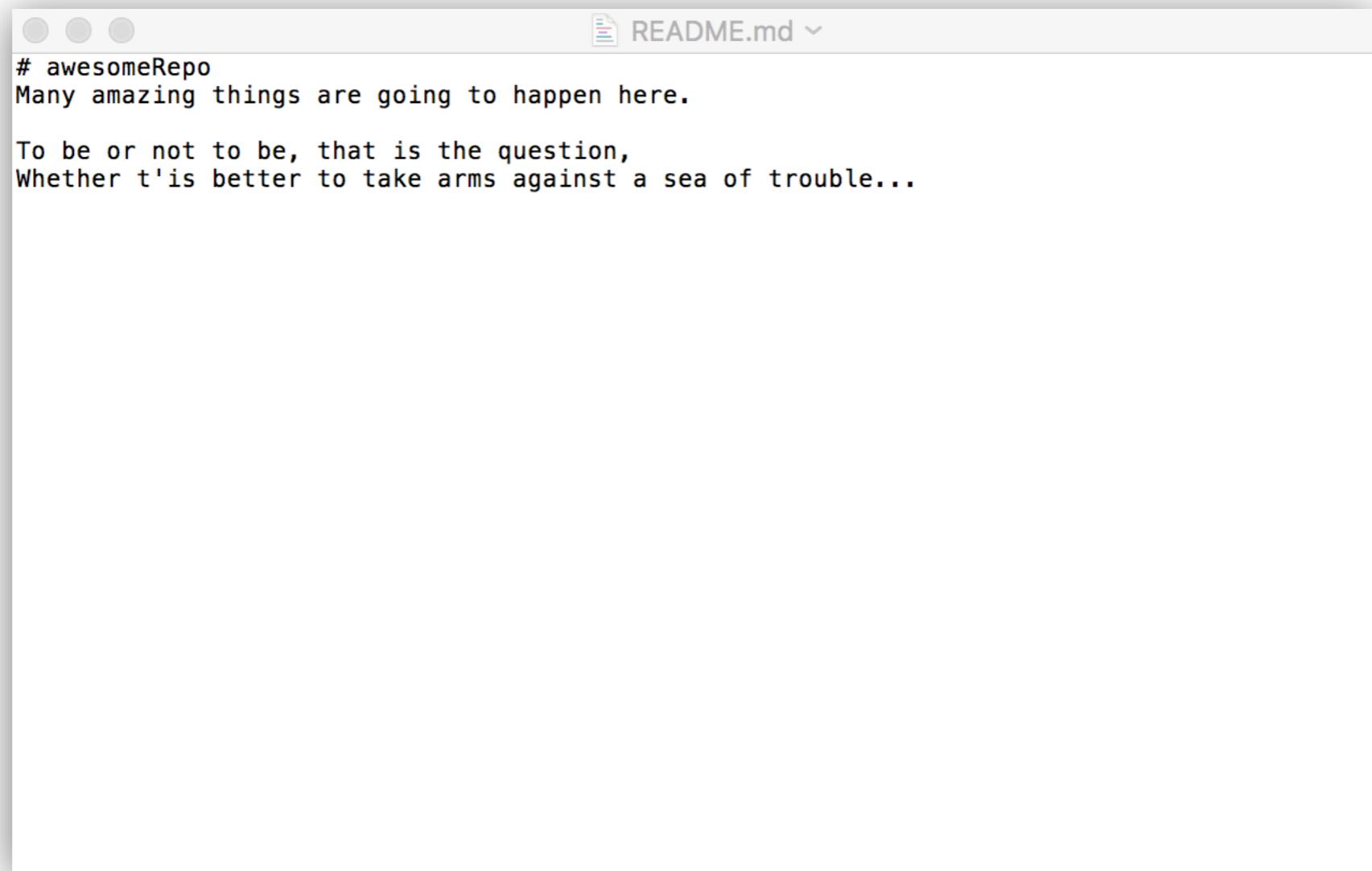
```
Last login: Thu Jun  7 17:12:45 on ttys001
Erichs-MacBook-Pro:~ erich$
```

Simple workflow for getting started



```
Last login: Thu Jun  7 17:12:45 on ttys001
Erichs-MacBook-Pro:~ erich$
```

Simple workflow for getting started



Simple workflow for getting started

```
awesomRepo — bash — 110x50

Last login: Thu Jun  7 17:12:45 on ttys001
[erichs-MacBook-Pro:~ erich$ git clone git@github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
Cloning into 'awesomeRepo'...
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), 12.51 KiB | 4.17 MiB/s, done.
[erichs-MacBook-Pro:~ erich$ cd awesomeRepo/
[erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
[erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
[erichs-MacBook-Pro:awesomeRepo erich$ git commit -a -m "For some reason, I really feel this sentence rings"
[master e8d83ab] For some reason, I really feel this sentence rings
 1 file changed, 4 insertions(+), 1 deletion(-)
[erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
[erichs-MacBook-Pro:awesomeRepo erich$ git push
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 442 bytes | 442.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
  7e55bee..e8d83ab  master -> master
[erichs-MacBook-Pro:awesomeRepo erich$ ]
```

Simple workflow for getting started

```
awesomRepo — bash — 110x50

Last login: Thu Jun  7 17:12:45 on ttys001
[erichs-MacBook-Pro:~ erich$ git clone git@github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
Cloning into 'awesomeRepo'...
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), 12.51 KiB | 4.17 MiB/s, done.
[erichs-MacBook-Pro:~ erich$ cd awesomeRepo/
[erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
[erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
[erichs-MacBook-Pro:awesomeRepo erich$ git commit -a -m "For some reason, I really feel this sentence rings"
[master e8d83ab] For some reason, I really feel this sentence rings
 1 file changed, 4 insertions(+), 1 deletion(-)
[erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
[erichs-MacBook-Pro:awesomeRepo erich$ git push
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 442 bytes | 442.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
  7e55bee..e8d83ab  master -> master
[erichs-MacBook-Pro:awesomeRepo erich$ ]
```



Simple workflow for getting started

```
awesomRepo — bash — 110x50

Last login: Thu Jun  7 17:12:45 on ttys001
[Erichs-MacBook-Pro:~ erich$ git clone git@github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
Cloning into 'awesomeRepo'...
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), 12.51 KiB | 4.17 MiB/s, done.
[Erichs-MacBook-Pro:~ erich$ cd awesomeRepo/
[Erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
[Erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
[Erichs-MacBook-Pro:awesomeRepo erich$ git commit -a -m "For some reason, I really feel this sentence rings"
[master e8d83ab] For some reason, I really feel this sentence rings
 1 file changed, 4 insertions(+), 1 deletion(-)
[Erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
[Erichs-MacBook-Pro:awesomeRepo erich$ git push
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 442 bytes | 442.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
  7e55bee..e8d83ab  master -> master
[Erichs-MacBook-Pro:awesomeRepo erich$ ]
```



Simple workflow for getting started

```
awesomRepo — bash — 110x50

Last login: Thu Jun  7 17:12:45 on ttys001
[Erichs-MacBook-Pro:~ erich$ git clone git@github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
Cloning into 'awesomeRepo'...
remote: Counting objects: 4, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), 12.51 KiB | 4.17 MiB/s, done.
[Erichs-MacBook-Pro:~ erich$ cd awesomeRepo/
[Erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
[Erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
[Erichs-MacBook-Pro:awesomeRepo erich$ git commit -a -m "For some reason, I really feel this sentence rings"
[master e8d83ab] For some reason, I really feel this sentence rings
 1 file changed, 4 insertions(+), 1 deletion(-)
[Erichs-MacBook-Pro:awesomeRepo erich$ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
[Erichs-MacBook-Pro:awesomeRepo erich$ git push
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 442 bytes | 442.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To github.com:Duke-Translational-Bioinformatics/awesomeRepo.git
  7e55bee..e8d83ab  master -> master
[Erichs-MacBook-Pro:awesomeRepo erich$ ]
```