Introduction to **Git** and **Github**

By Athol Whitten for Super Advanced R (FISH512)

University of Washington (SAFS), June 2014

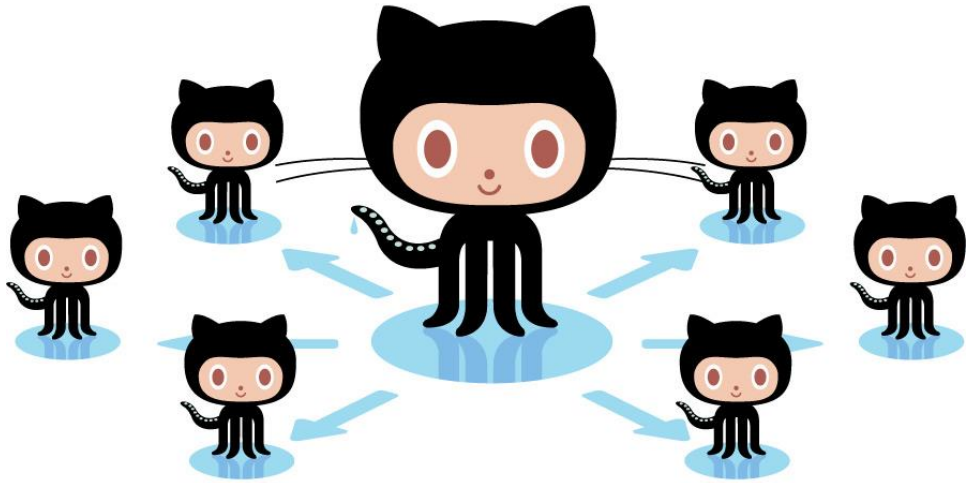
What is **Git**?

- Git is a distributed version control system
- Written by Linus Torvalds
- Designed to handle both large and small projects
- Allows tracking, branching, merging, and multiple workflows

Click for more git info

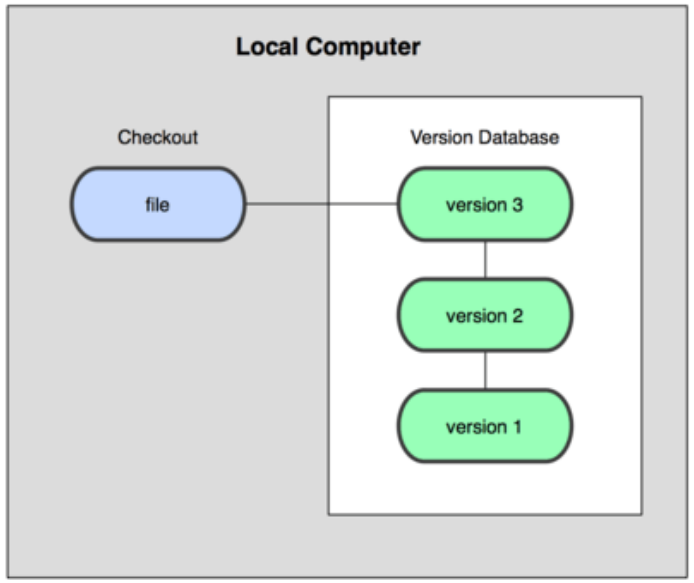
What is **Github**?

- Online host for **Git** repositories
- Allows for '**Social Coding**'
- Free for public projects
  *private repositories possible too
- Acts as server for **collaborative** projects

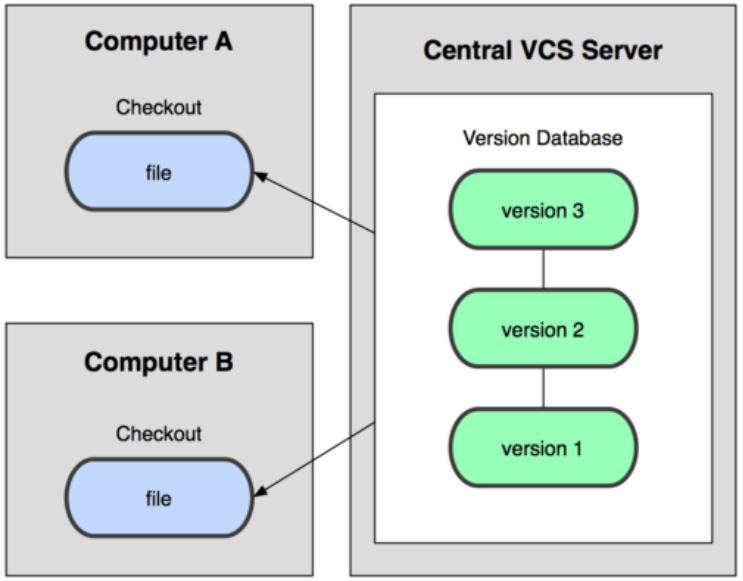


About **version control**

- Version control records changes to a file or set of files over time

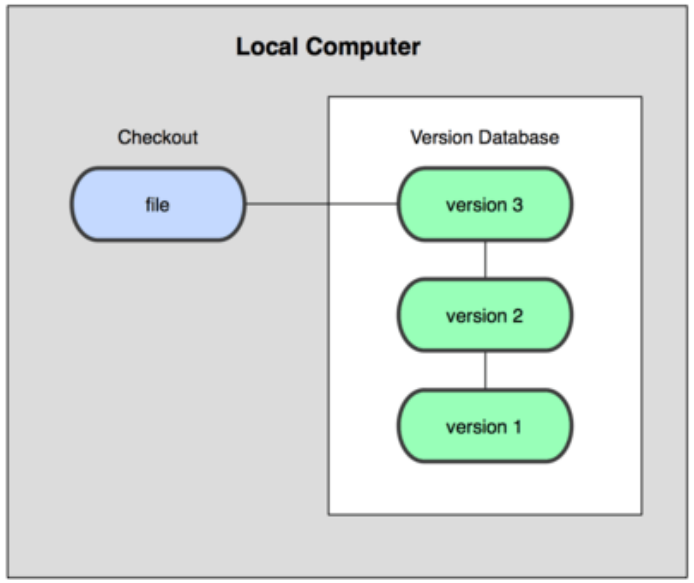


Local System

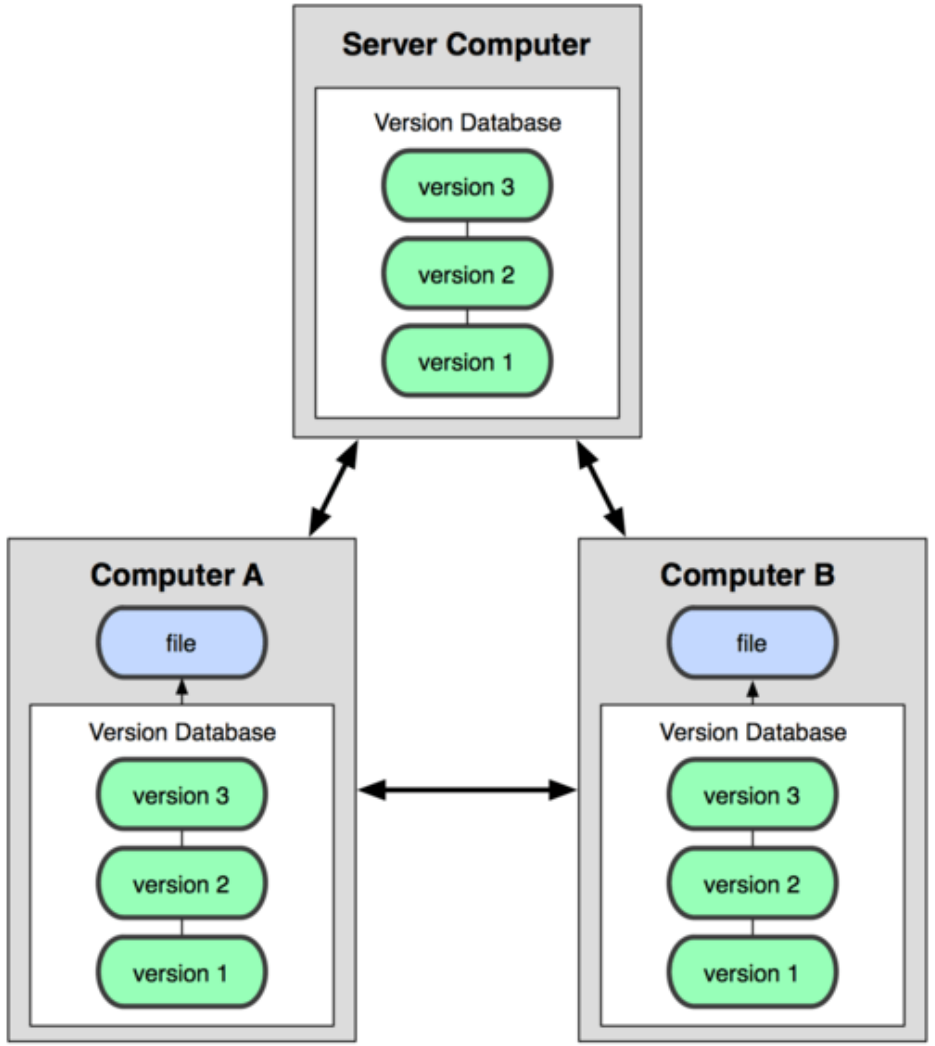


Centralized Version Control

About **version control**



Local System



Distributed Version Control

Why **Git**?

- Free and open source

- Many features
- Simple (**for the basics**)
- Small footprint (**single directory**)
- A suite of interrelated tools
- Offline version control
- Platform agnostic
- Branching based workflow

Getting **started**

- **Git** can be downloaded from http://git-scm.com/
- It’s easy to install, only binaries are required
- Put Git on your PATH (or equivalent)
- Run from Command, Shell, BASH, etc.

Setting up

- Test current version of git [`git --version`]

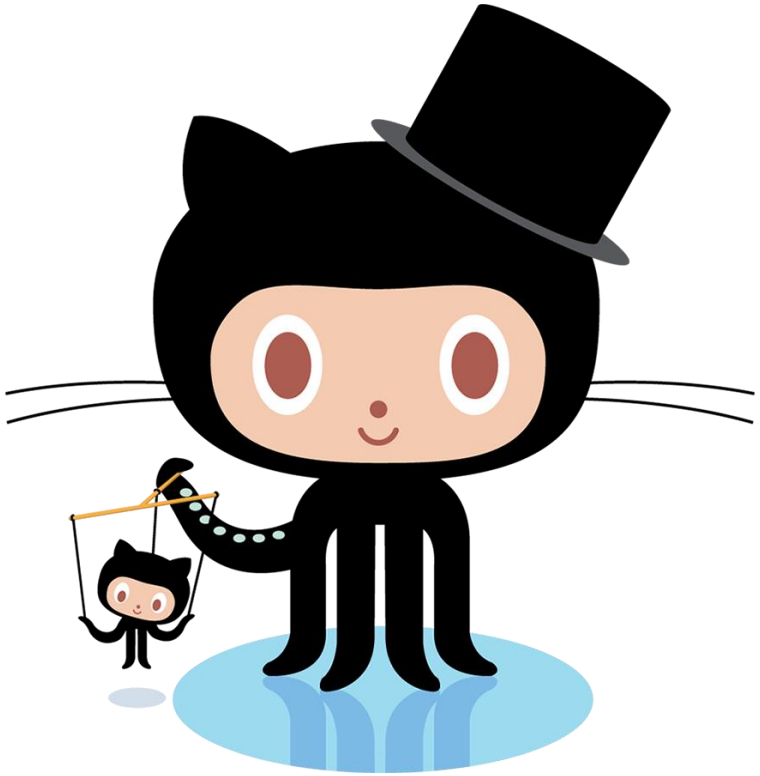
- Configure your Git [`git config`]

```
$ git config --global -l
```

- Set your name, email address, and editor:

```
$ git config --global user.name ‘Your Name’
$ git config --global user.email ‘you@web.com’
$ git config --global core.editor ‘notepad’
```

The **basics**



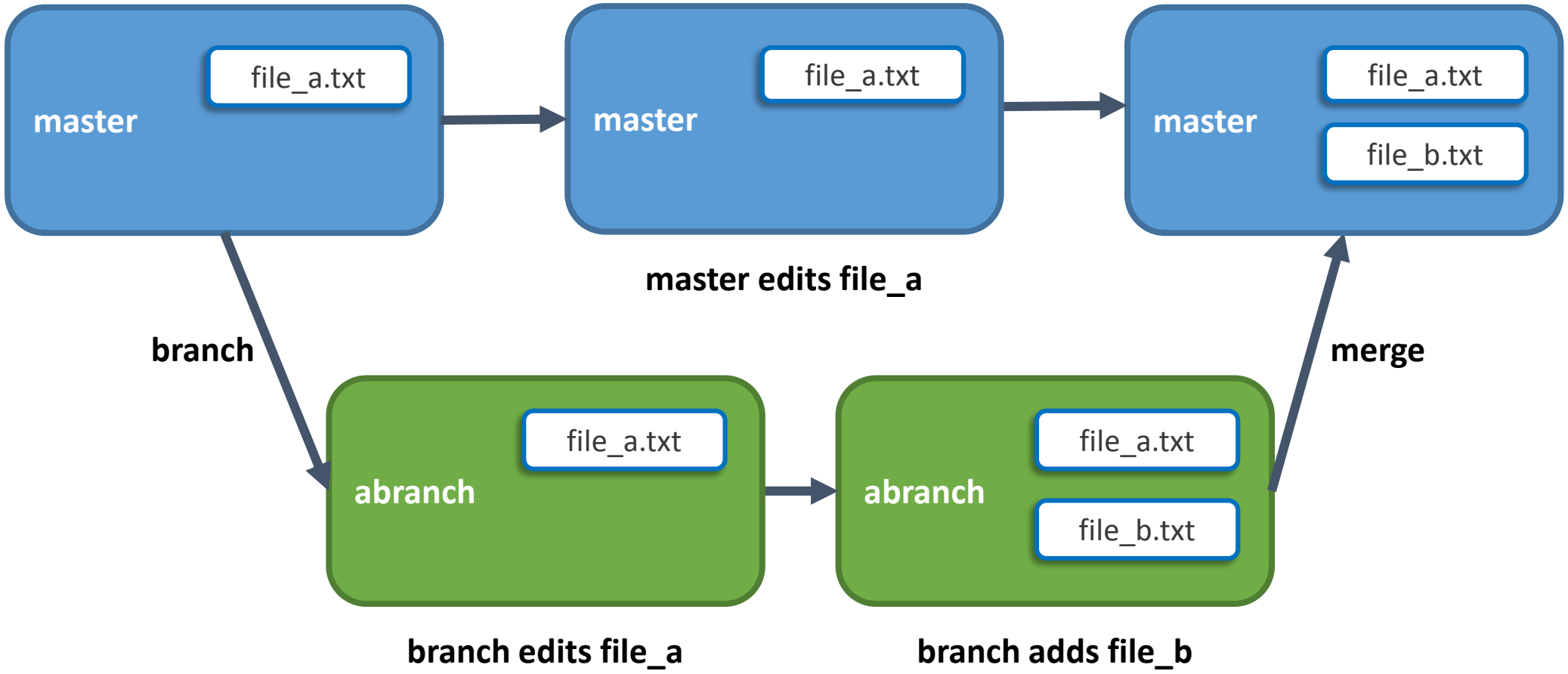
- Initialize a repository [`git init`]
- Stage changes [`git add`]
- View changes [`git status`] or [`git diff`]
- Commit changes [`git commit`]
- Add multiple files with wildcards

```
git add '*.txt'
```

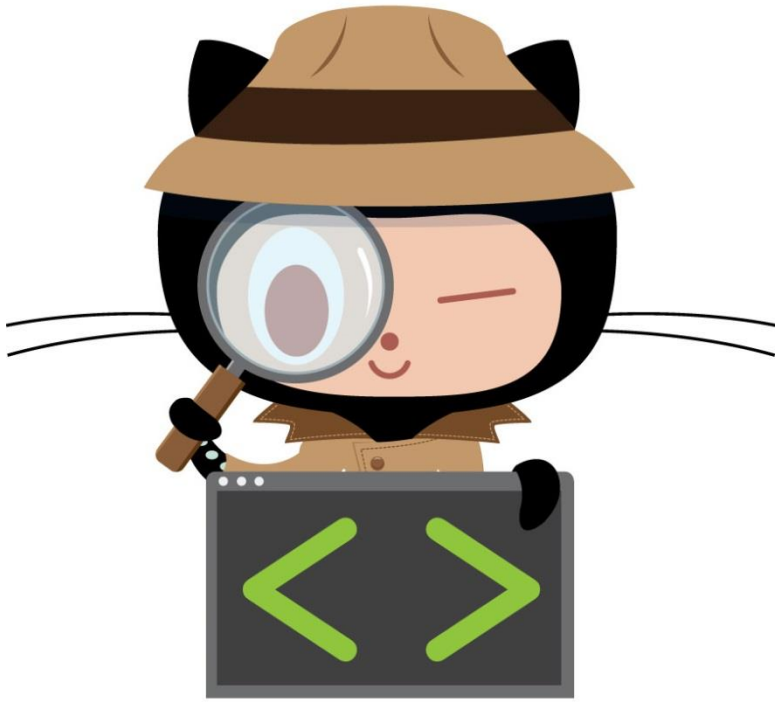
- Commit with an inline message

```
git commit -m 'inline message here'
```

Git branching



Branching, merging, tracking



- View the change log [`git log`]

  * each commit has a unique number (**sha**)

- Branch a repo [`git branch`]
- Checkout the branch [`git checkout`]
- Make changes, then `add` and `commit`
- Merge the changes [`git merge`]
- Try [`git tag`] to mark a special point

Exercise: Add, Commit, Branch, Merge

- Try this typical workflow example:

1. Make some changes to your files
2. `add` changes
3. `commit` changes
4. `branch` repository
5. `checkout` new branch
    - make changes to new branch files
6. view the changes to your files: `diff`
7. `merge` branch to master

Getting **started**

- Create a Github account at https://github.com

  - Create a new **repository** (leave it blank)

  - It's a good idea to give your repository and project the same name

- Add a remote [git remote]
  - `$ git remote add origin https://github.com/name/project.git`

- Push a repository [git push]

```
$ git push –u origin master
```

name = github account name

project = name of your project and/or repository

Exercise: Local changes, push to **Github**

- Try this typical **Github** workflow:

1. make changes to a repository
2. add changes
3. commit changes
4. push changes
5. make new branch
6. make changes in branch
7. push new branch
8. merge changes locally
9. push merged changes

- **Hint**: Work on the README file to see changes via Github on the main page of your repository

Collaborating via **Github**

- Clone a repository [`git clone`]
- Fork a repository [`github equivalent`]
- Work as **collaborators** or **fork**
- Having made local changes, you can `push` (with permission), or make a `pull request` to the repository owner

Exercise: Fork, Add, Commit, Pull Request

- Try this typical **collaborative** workflow example:

1. `fork` a repository
2. `clone` to local machine
3. make changes
4. `add` changes
5. `commit` changes
6. `push` to forked repository
7. `submit` pull request

More information

- http://git-scm.com/about for introduction to **Git**
- http://git-scm.com/book for the nitty gritty
- http://gitref.org for a basic how-to guide
- https://github.com/ for overview of **Github**
- https://try.github.io for an interactive trial