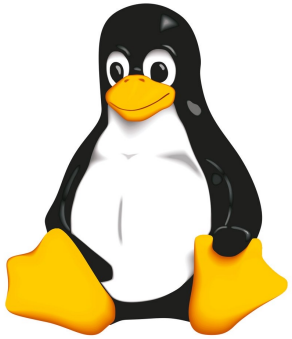
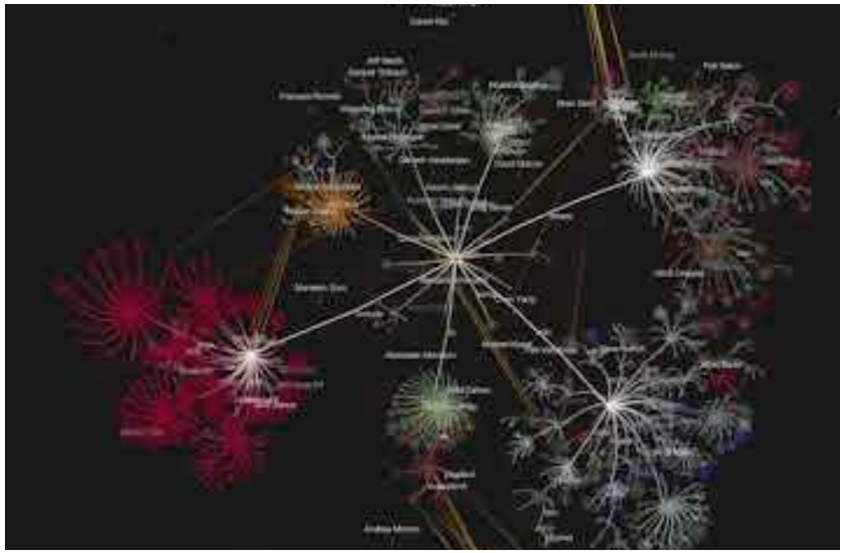
Autonomous Systems – Lab 1

Christoph Killing

In todays lab …

… penguins and fireworks





Preliminaries – Lab Sessions

- Tools necessary for course (i.e. Linux, git, ROS, Unity …)
- Introduction of homework assignments

Preliminaries – Course Organisation

- HW 1, 2
  - independent work to get going and check setup
- From HW 3 onwards
  - group projects
  - Each group is required to submit every homework to be eligible to take part in the project
  - Final grade: 100% final project (in groups; this will be big!)
  - More details to follow

- Please use the Moodle Forum if you want help fast

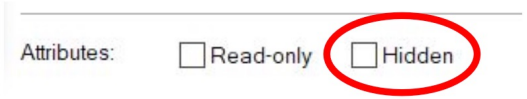
Linux

“A computer is like air conditioning: it becomes useless when you open Windows’

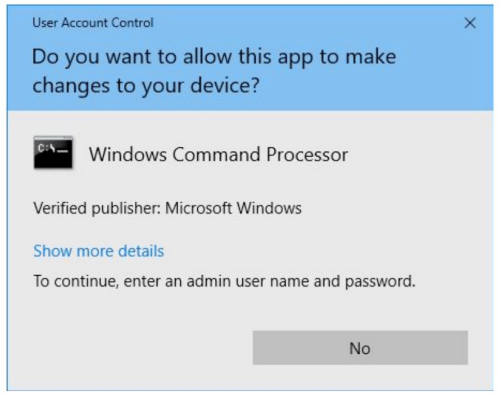
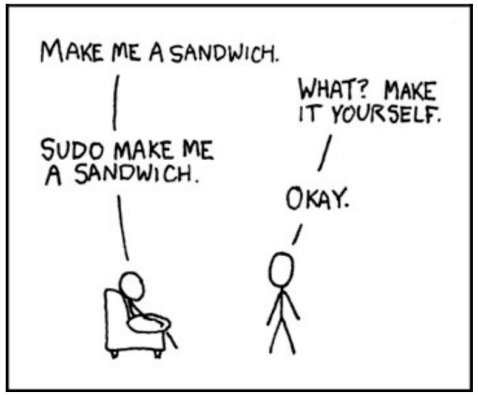
Linus Torvalds

- Family of open-source operating systems
- Most devloper-friendly OS
- Ubuntu is most commonly used distribution
  - Stable Ubuntu 18.04 LTS “long-term support”
- Full ROS-compatibility
  - Robot Operating System

Linux vs. Windows

ROOT	`C:\`	→	`/`
HOME	`C:\Users\<username>`	→	`/home/<username>` OR `~/`
CASE	CaSe INsenSITIVE "a" = "A"	→	Case Sensitive "a" ≠ "A"
HIDDEN		→	Name begins with "."
EXECUTABLE	`.exe` format	→	No extension

Linux – sudo command

HOME	`C:\Users\<username>`	→	`~/` - **sudo NOT required**
ROOT	`C:\`	→	`/` - **sudo required**
USER PRIVILEGES		→	

"Run as Administrator"

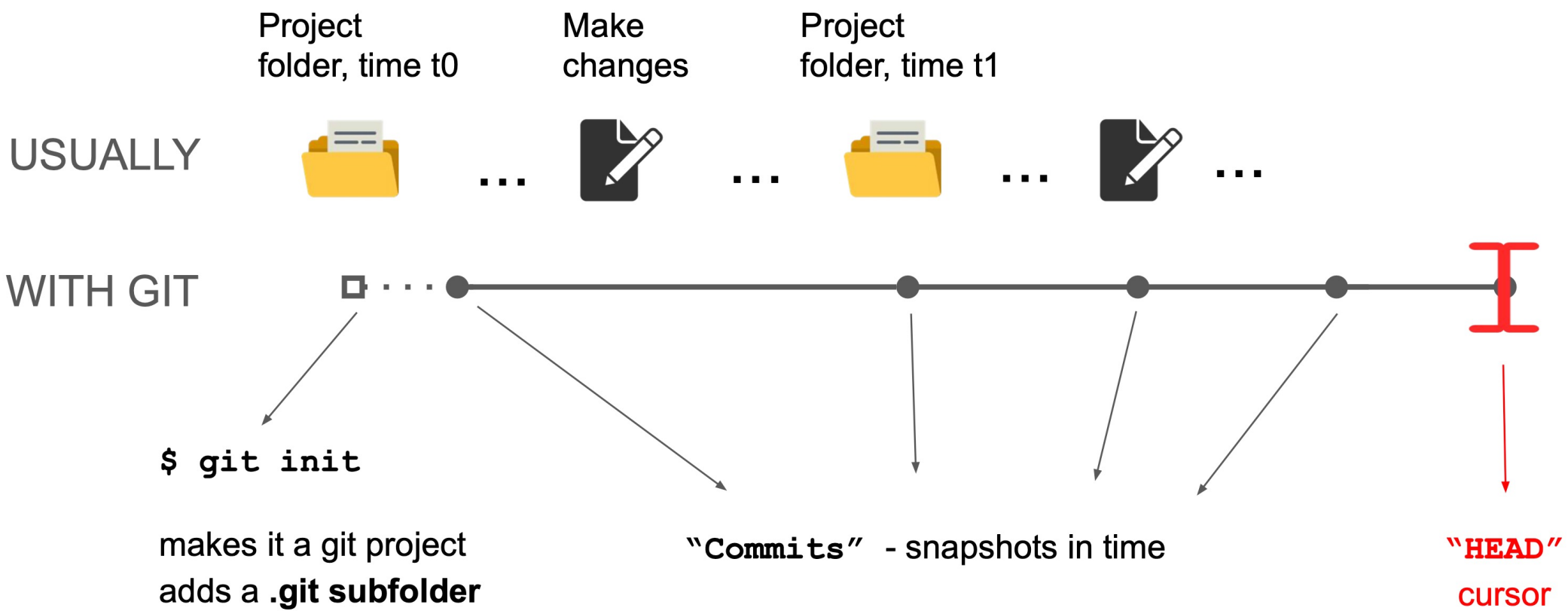
The "**sudo**" command
(a.k.a. SUperuser DO)

Linux – Terminal

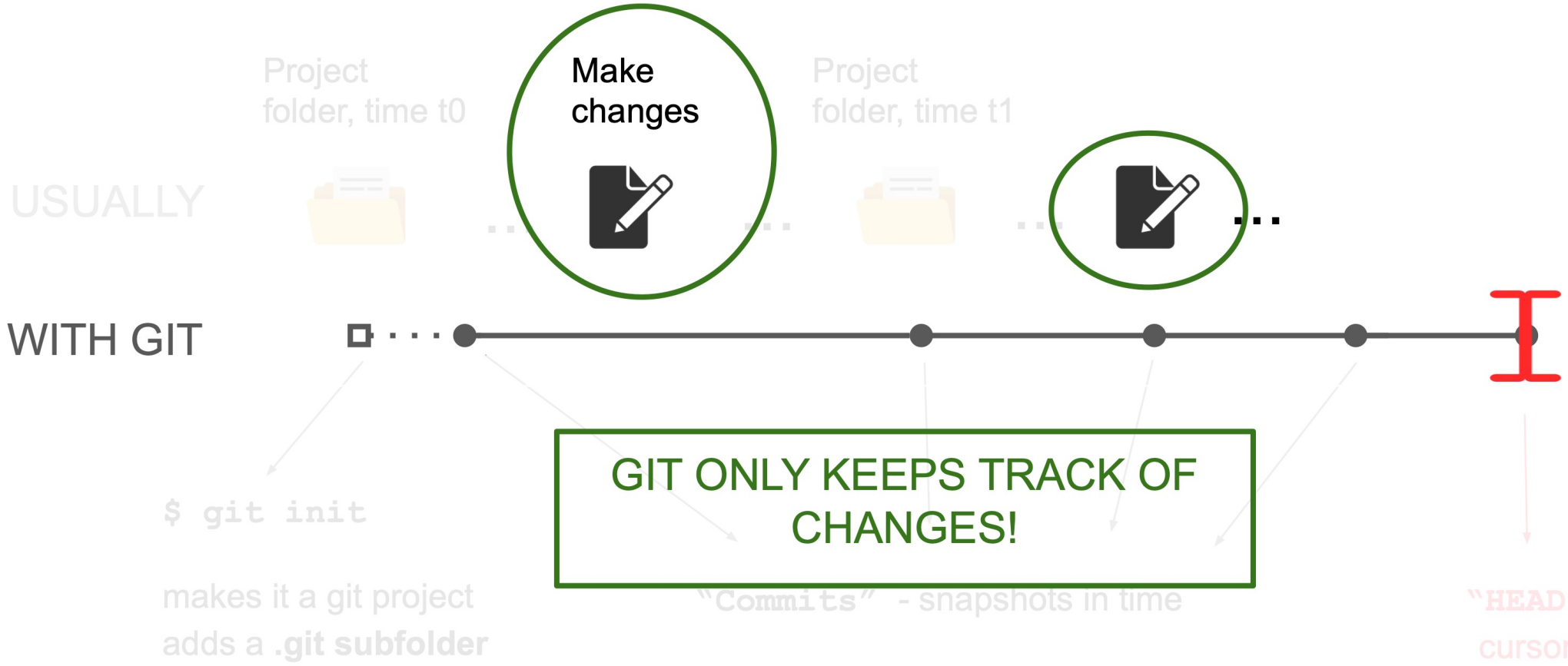
<UserName>@<PCname>:<WorkingDir>$

Open a Terminal window	CTRL + T
Print Working Directory:	**pwd**
Change directory:	**cd, cd .., cd <path>**
Make directory:	**mkdir <dir_name>**
Create file:	**touch <file_name>**
Move/rename file:	**mv <file_name>**
Edit text file:	**gedit <file_name>**
List dir contents:	**ls, ls -a** (for hidden files)
Delete file / dir:	**rm <file_name>,** **rm -r <dir_name>**
Autocompletion:	**<TAB key>**
Install from package manager:	**apt-get**
Quick reference / manual pages:	**whatis <cmd>, man <cmd>**
Run an executable:	**./<executable_name>**

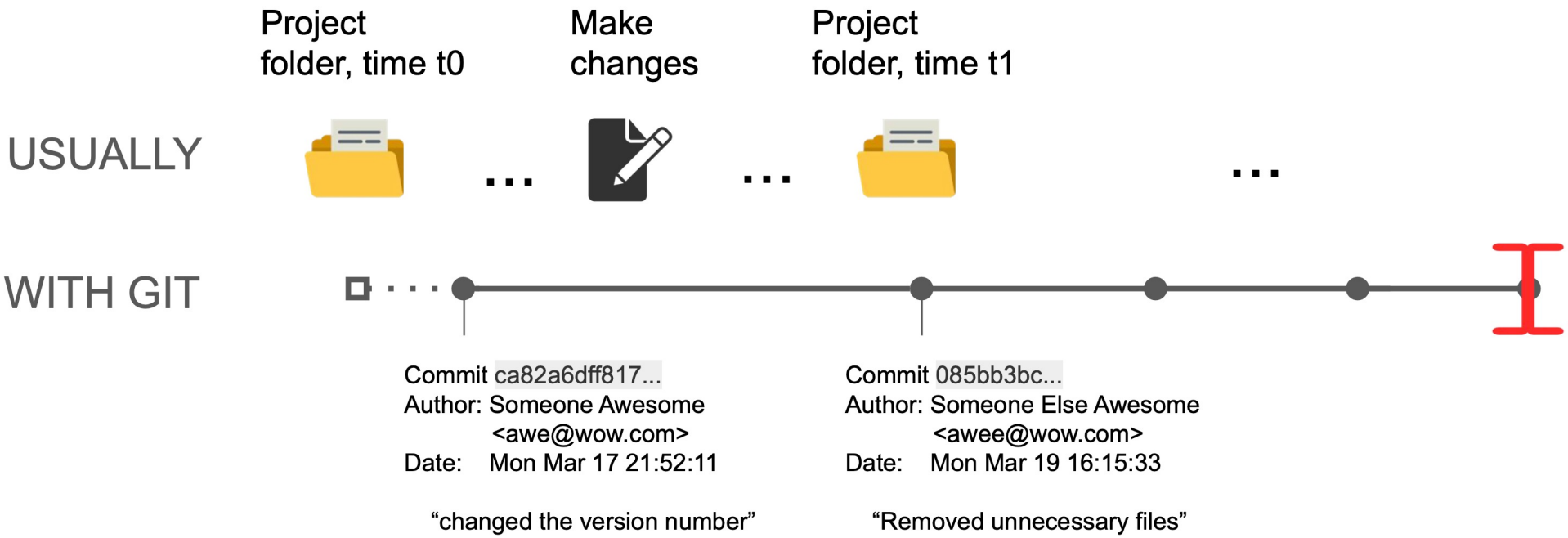
Git – Version Control



Git – Version Control



Git – Version Control



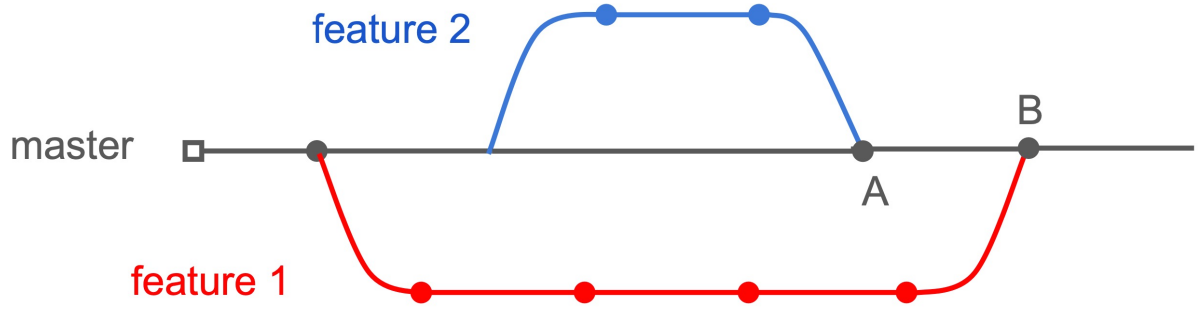

```
$ git checkout 085bb3
```

> Moves the head to a given commit
(all files are rolled back to that point in time)

Git – Basic Commands

- Setup:
  - git clone [link to your repo]
  - git status
- Most basic commands:
  - git pull
  - git add [i.e. file.txt]
  - git commit –m “[your commit message]”
  - git push

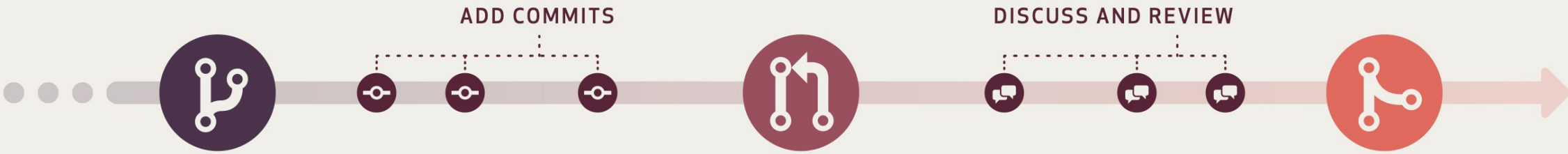
Git - Branching



- Git allows you to keep several branches of your code
- Branch-out from master
- Merge back into master
- Careful! Merge-Conflicts can arise at points A, B

WORK FAST
WORK SMART
THE GITHUB FLOW

The GitHub Flow is a lightweight, branch-based workflow that's great for teams and projects with regular deployments. Find this and other guides at **http://guides.github.com/**.



CREATE A BRANCH

Create a branch in your project where you can safely experiment and make changes.

OPEN A PULL REQUEST

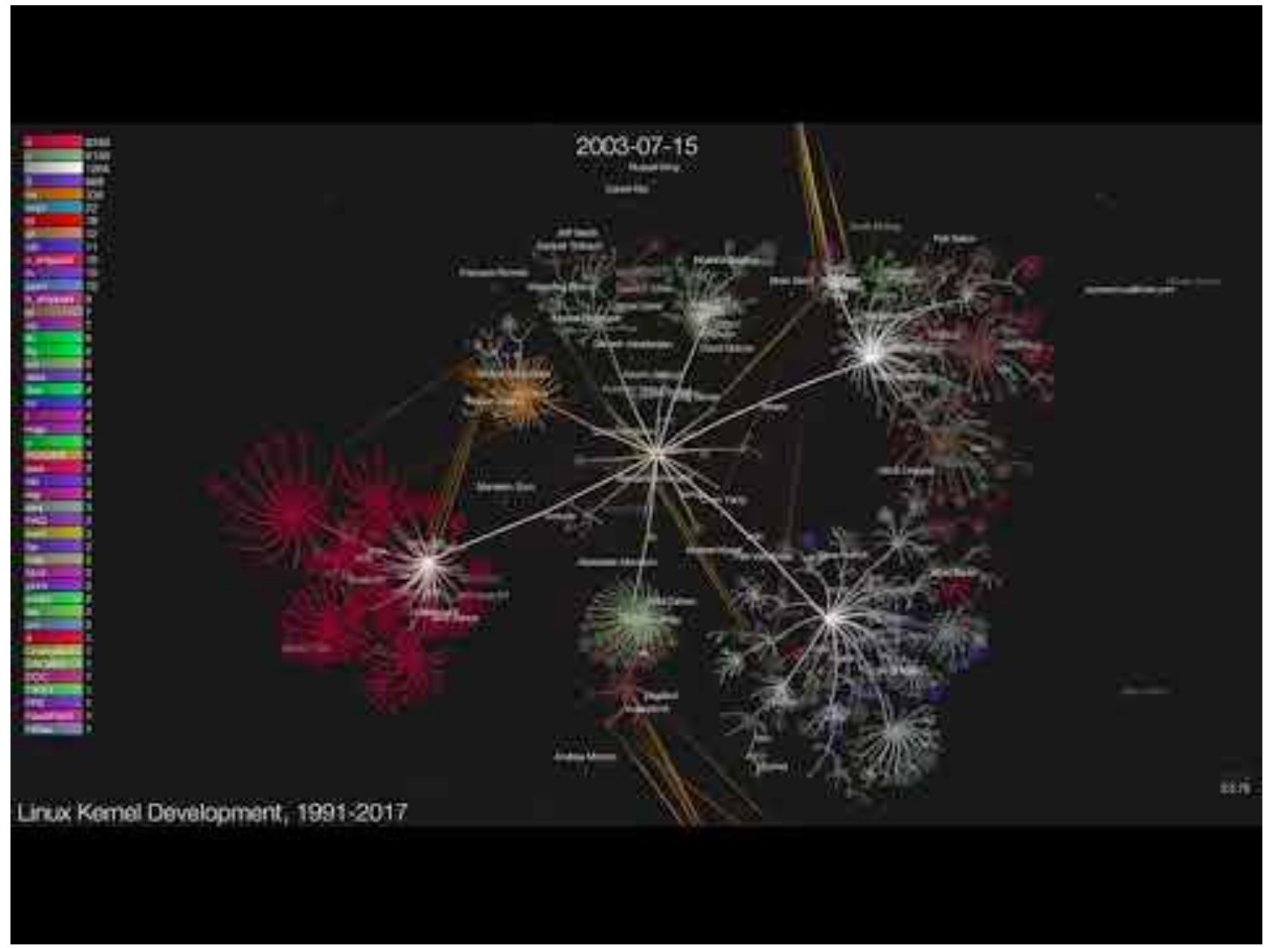
Use a pull request to get feedback on your changes from people down the hall or ten time zones away.

MERGE AND DEPLOY

Merge your changes into your master branch and deploy your code.

https://guides.github.com/introduction/flow/

Git



Linux kernel development 1991 - 2017, Gource

Homework

1. Create your own GitLab Repo
2. Clone the course GitLab
   a) Code templates
   b) Slides
   c) Scripts
3. Solve simple C++ exercise
4. Start forming groups
   a) Use Moodle to find teammates
   b) Details can be found in Homework description

Terminal: git pull →