AMS 129

Youngjun Lee

Welcome to AMS 129

- Youngjun Lee
  - PhD Candidate, Applied Mathematics
- Email: ylee109@ucsc.edu

- Office Hours
  - TBD
  - or by appointment

Welcome to AMS 129

- Lectures
  - MWF 09:20AM -- 10:25AM
- Sections (TA: Sky Trigueiro)
  - Th 01:30PM -- 03:00PM
  - F 11:00AM -- 12:30PM
  - **NO lab sections in the first week**

Disclaimers

- This course is not a CS/CE course
  - NO software engineering
  - NO hardware architecture
- We will focus on **how** to use scientific tools

Grading

- Homework (60%); Final Project (40%)
- All homework and the final project are individual project.
  - **Do not share your code(s) with others.**
- **No late submission.**

Syllabus

- Week 1 -- 2
  - Linux basic commands
  - Version controls
- Week 3 -- 4
  - Fortran programming
  - Compiler, Makefile
  - I/O, Debugger

Syllabus

- Week 5 -- 8
  - Python programming
  - Sphinx
  - Jupyter notebook, NumPy, Pandas, Matplotlib
- Week 9 -- 10
  - C programming
  - Other useful scientific computing tools

Googling

- Googling is highly recommended in this course
  - But please do not just copy the solutions.
    - Know what you are doing
- YouTube is another great place to find some tutorials

Linux

- This course is premised on UNIX-like computers
  - Linux (Ubuntu, Fedora, openSUSE, ...)
  - macOS

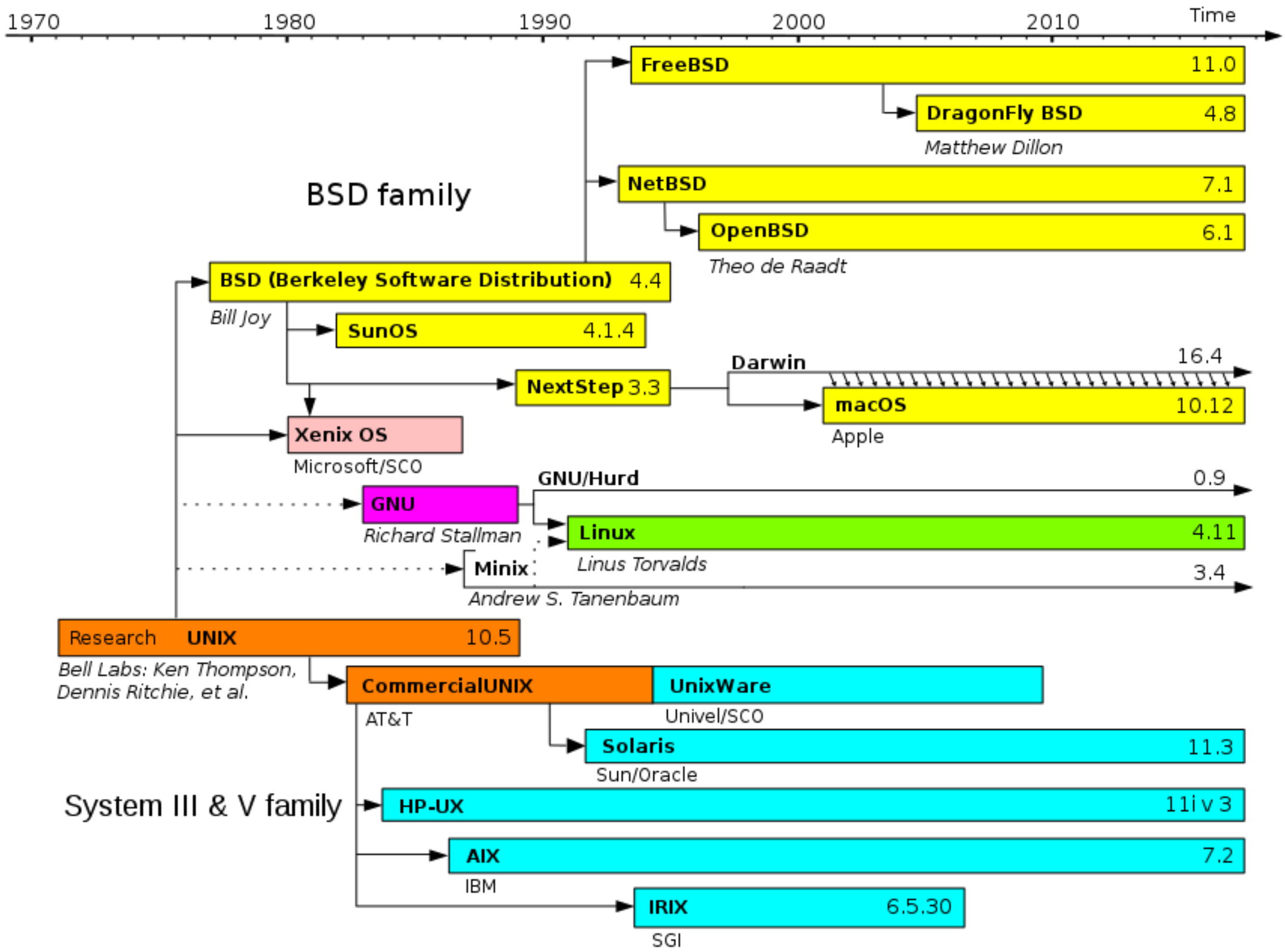


Image from Wikipedia.org

Linux

- If you have only Windows machine, you have couple of options
  - Install Linux on separate drive and dual boot
    - Or you can use USB drive stick but unstable
  - Install Linux on virtual machine
    - VirtualBox (https://www.virtualbox.org) is free
- Use Cygwin (https://www.cygwin.com) - not recommended
- Use campus resources and store the file personally

Package Managers

- Install, Remove (mostly) command line programs
- Most Linux systems have their own package managers
  - **apt** for Ubuntu
  - **dnf** for Fedora
- If you are using macOS you need to install either
  - Homebrew (https://brew.sh) - recommended
  - Macport (https://www.macports.org)

Shell Commands

- man
  - Move: j, k or ctrl+d, ctrl+u
  - Exit: q
- ls
- mv
- cp
- rm
- cat

Shell Commands

- mkdir
- cd
- touch
- find
- grep

Operators

- Redirecting
  - > : redirect the output to a new file
  - >> : redirect the output to a existing file and append it
- Piping
  - | : feed the **output of left** as input to **program of right**
    - ls | grep [pattern]
    - ls | sort

Operators

- Multiple line commands
  - A; B : run A and then B, regardless of success of A
  - A && B : run B if A succeeded
  - A || B : run B if A failed

Bash

- Bash is an Unix Shell program
- If you opened terminal on macOS or Linux, it is (mostly) a Bash
- You can choose either
  - Bash
  - zsh
  - csh
  - ...

Bash

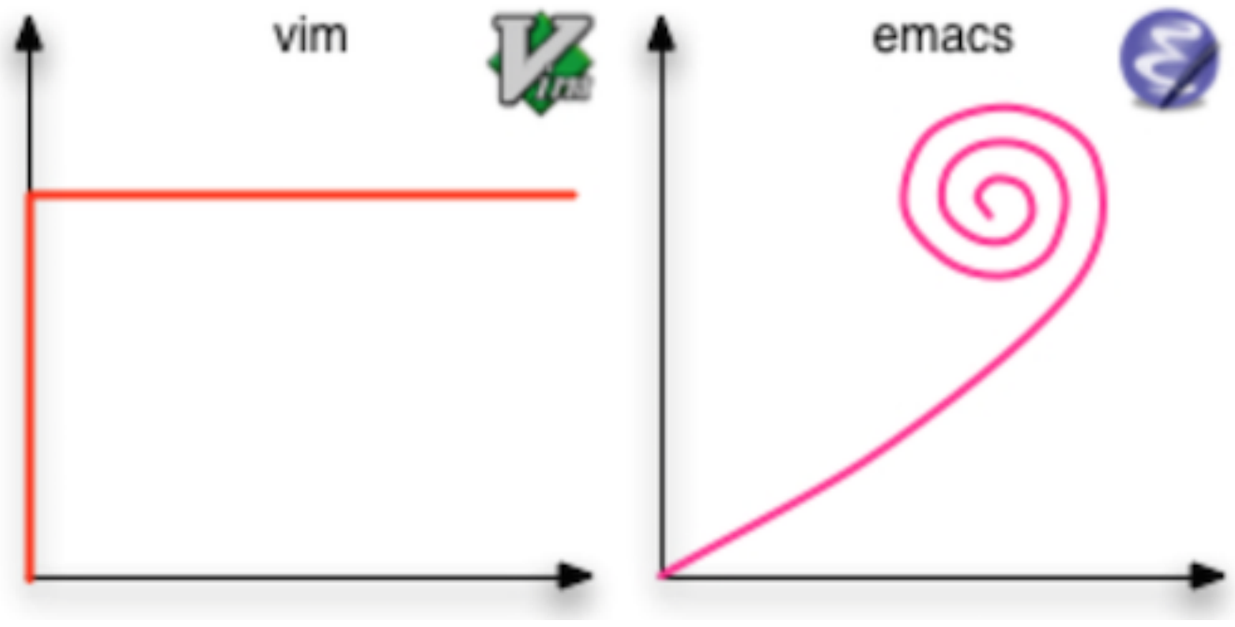
- Environment variables for Bash
  - HOME
    - indicate home directory
  - PATH
    - indicate search path for commands
  - USER
    - current user name

Bash

- Bash configuration file (.bashrc) is stored in your home directory
  - cd ~ (equiv. cd $HOME)
  - vim .bashrc
- Wait.. vim?

Editor

- Two most popular (command line) editors
  - vim (or vi)
  - Emacs



Editor

- Yes, you can use modern, graphical editor such as
  - sublime text (https://www.sublimetext.com)
  - atom (https://atom.io)
  - visual studio code (https://code.visualstudio.com)
- But.. you need to use vim/Emacs when you are doing something remotely
  - ssh ylee109@sftp.ucsc.edu

Bash

- .bashrc file is a script that running every time you launch the Bash
- alias
  - alias 'll'='ls -al'
  - alias '..'='cd ..'
- export
  - export PATH=/usr/local/bin:$PATH

Bash

- A lot of things could be easier with modifying .bashrc
- You can find some great examples of .bashrc
  - github.com

Version Control

- Case 1
    - You are involving a large project with many others. You and your teammates have to modify a single project at once.
- Case 2
    - You have your own project, but you have to work in many different places.
- Case 3
    - The project need to be published in many different versions.

git & svn

- git
  - The most popular version control solution
  - Invented by a Linus Torvalds to control Linux project
  - Decentralized
- svn
  - Formally used solution
  - Centralized

git

- git init
- git status
- git add
- git commit
- git log
- git diff

git

- You can add remote server to store the file online (or cloud)
  - git remote add [nickname] [address]
- If you want to see the remote server
  - git remote -v

git

- Three options for git server
  - github.com
    - Non-private server for free accounts
  - bitbucket.com
    - Private server for free accounts
  - MakeYourOwnServer

git

- git pull
- git push

git

- If you want to revert ALL changes in LOCAL
- git fetch origin master && git reset --hard

git

- git clone
- from server

Remote System

- You can login to another system.
- ssh [YourID]@[ServerAddress]
- eg) ssh [CruzID]@sftp.ucsc.edu (with Blue password)

Remote System

- If you want to copy a file,
- scp [File] [YourID]@[ServerAddress]

- eg) scp ./.bashrc ylee109@sftp.ucsc.edu:~/