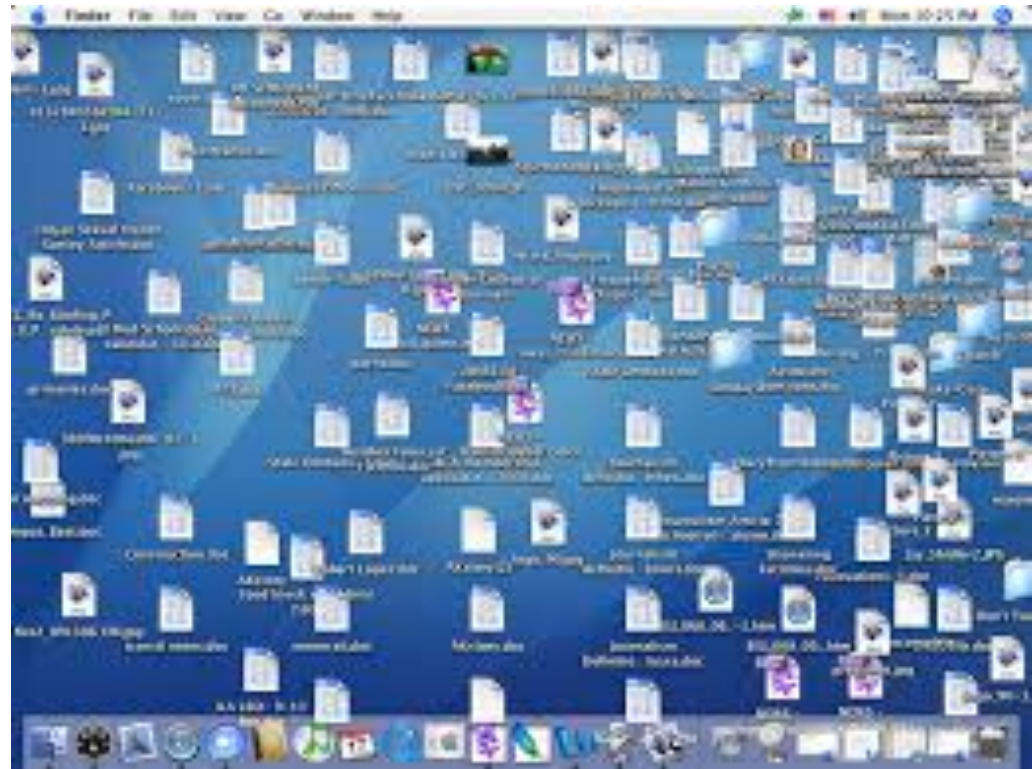


Class 3

Project Directories

Organization of files

- In the course of a project, you will have
 - Data files
 - Processed data files
 - Intermediate analysis files
 - Final output files
 - Images
 - Scripts
 - Notes
 - Commands



No!

Organization of files

- Essential for:
 - Reproducing or rerunning results
 - Sharing with other people
 - Automating tasks
 - Managing a large project with many files
 - Your sanity

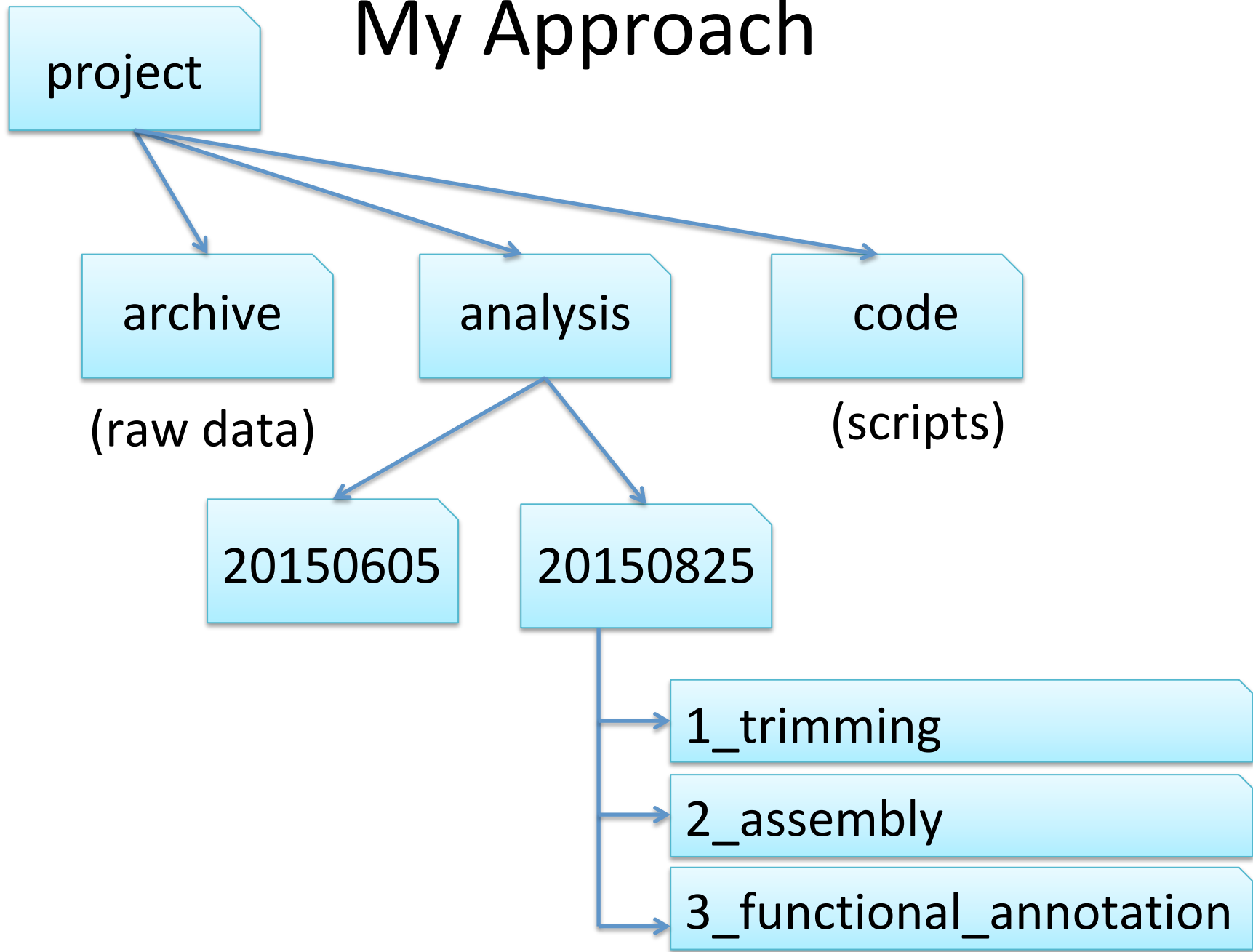
To start

- Everything for one project will go in one master folder
- Give this folder a sensible, descriptive name

The organization inside this folder varies from lab to lab. One strategy is outlined in Noble (2009) “A Quick Guide to Organizing Computational Biology Projects”


Just be consistent.

My Approach



More Best Practices

- Use sensible, descriptive names for all folders and files
- No spaces
- Use file extensions (.txt, .fasta, .sam)
- Use leading zeros
 - File001.txt
 - File002.txt
 - ...
 - File192.txt



These will display
in the correct
order

Documentation

Simplest
method:
README
file

- Methods and workflows
 - Write down your full command lines
 - Preferably in a script
 - Software versions
- Origin of data
 - Where did it come from? Who gave it to you?
 - This includes not just your experimental data, but any data you use – a reference genome, a set of genes, a database
 - Date of download
 - Version



File Permissions

With great power comes great responsibility

One of these deletes the tmp-data directory and any files that start with “aligned-reads”. The other deletes EVERYTHING in the current directory.

```
$ rm -rf tmp-data/ aligned-reads*  
$ rm -rf tmp-data/ aligned-reads *
```

Undo is not an available shell command. Get smart about protecting your data.

Protect your data

- Backups for your own computer
 - External hard drive
 - Mac – Time Machine
 - Windows – Backup and Restore
 - Online backup software that is automated
 - iDrive, BackBlaze, Carbonite, etc
 - Costs money



<https://www.techopedia.com/2/29440/trends/cloud-computing/cloud-vs-local-backup-which-do-you-need>



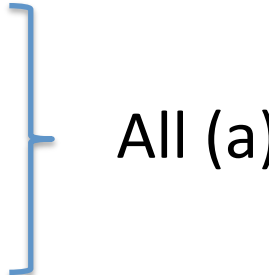
Protect your data

- Backups of data
 - External hard drive – these do fail, consider using 2
 - ACF backs up home directories but not project directories
 - Low likelihood of loss unless a major disaster happens!
 - **Keep an eye on project end date – you may lose data. No emails are sent right now.**
- But what if you delete it?
- Linux Permissions
 - Can use these to prevent yourself or someone else from overwriting or deleting the data
 - Read-only permission



READ ONLY


Linux Permissions

- File permissions are broken down into three user types:
 - User/Owner (u)
 - Group (g)
 - World/Other (o)All (a)
- Permissions can control how each user type can interact with the file
 - Read (r)
 - Write (w)
 - Execute (x)

Linux Permissions

Viewing permissions with `ls -l`

A file with completely open permissions:

 `-rwxrwxrwx mstaton1 tug2222 file.txt`

 User Group World

User and group can read/write:

`-rw-rw---- mstaton1 tug2222 file.txt`

User and group and outsiders can read:

`-r--r--r-- mstaton1 tug2222 file.txt`

Linux Permissions

- Changing owner
`chown mestato file.txt`
- Changing owner + group
`chown mestato:statongroup file.txt`
- Removing write permission from user
`chmod u-w file.txt`
- Removing execute permission from everyone
`chmod a+x file.txt`