Project Organization

Childhood Cancer Data Lab

Sources!

All ideas come from somewhere.

These are some sources that have inspired us (and provided material directly)

- Vince Buffalo: Bioinformatics Data Skills
- Jenny Bryan: https://speakerdeck.com/jennybc/how-to-name-files
- Danielle Navarro: https://slides.djnavarro.net/project-structure

Why does project organization matter?

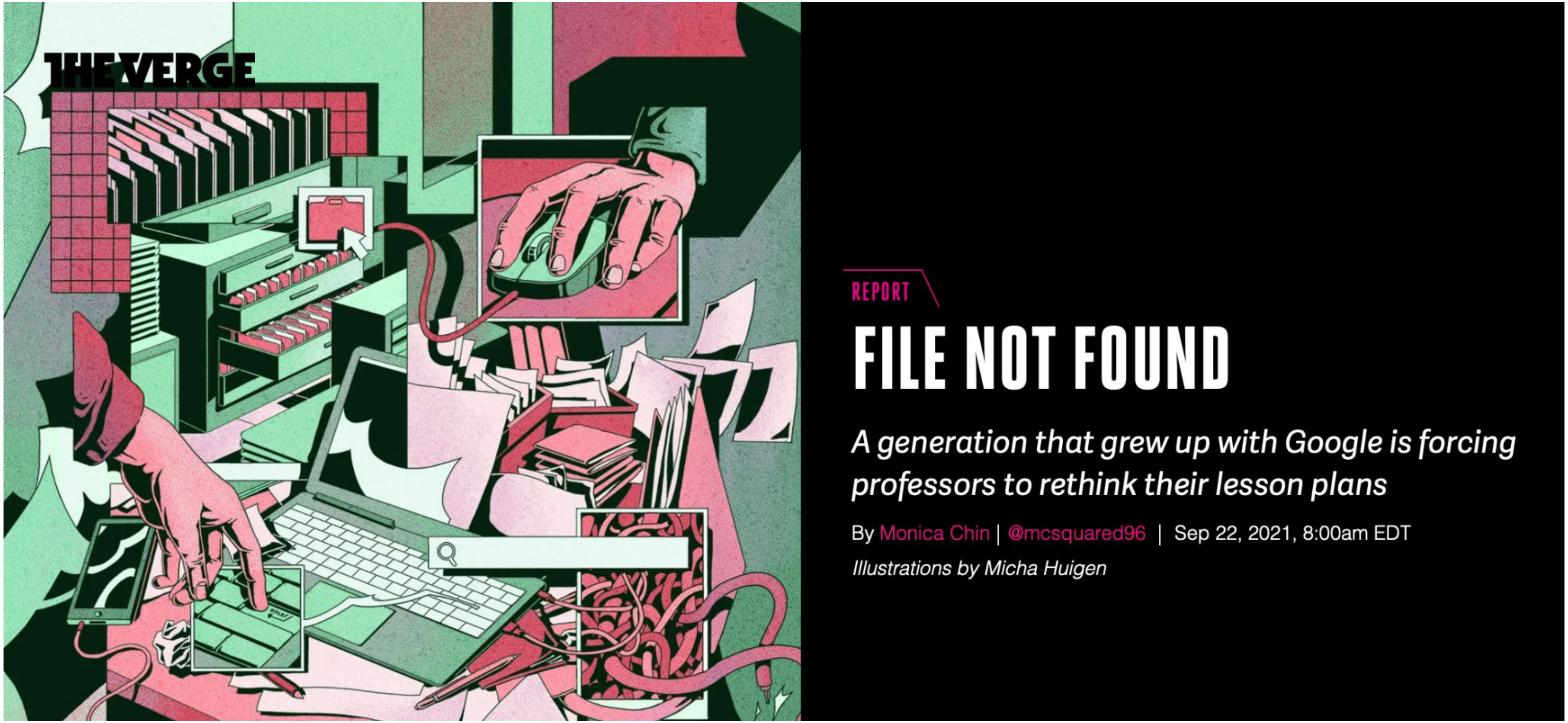
- Finding things takes a lot of time and effort
- Standard and predictable organization saves time
- Be kind to yourself & others!
  - Make your stuff discoverable
  - Follow consistent patterns

But I can just search!

- Google has trained us to search for content, and searching is great! Sometimes.

```
@071112_SLXA-EAS1_s_7:5:1:817:345
GGGTGATGGCCGCTGCCGATGGCGTCAAATCCCACC
+071112_SLXA-EAS1_s_7:5:1:817:345
IIIIIIIIIIIIIIIIIIIIIIIIIIIIII9IG9IC
@071112_SLXA-EAS1_s_7:5:1:801:338
GTTCAGGGATACGACGTTTGTATTTTAAGAATCTGA
+071112_SLXA-EAS1_s_7:5:1:801:338
IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII6IBI
```

- File names can be uninformative (we’ll come back to that!)
- Data files often don’t have searchable content
  - Even when they do, you might not know what to search for!
- **Metadata** describing the content might not be part of the file



REPORT

FILE NOT FOUND

A generation that grew up with Google is forcing professors to rethink their lesson plans

By Monica Chin | @mcsquared96 | Sep 22, 2021, 8:00am EDT

Illustrations by Micha Huigen

https://www.theverge.com/22684730/students-file-folder-directory-structure-education-gen-z

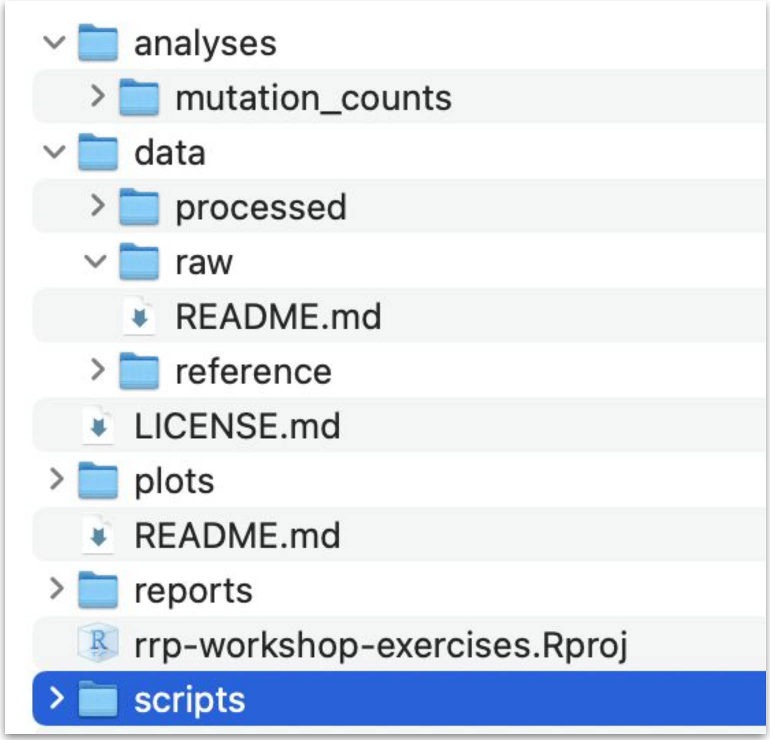
Where to start?

- **Use Folders/Directories!**

- Keep separate projects separated
- Separate sections for units within a project
  - Data
  - Code
  - Results
  - Reports

- **Documentation throughout!** ABD
  - Describe what files do and how they are organized

A typical project folder (for me!)



```
analyses
    mutation_counts
data
    processed
    raw
        README.md
    reference
LICENSE.md
plots
README.md
reports
rrp-workshop-exercises.Rproj
scripts
```

You'll see more of this project later!

The `data` folder

- This is where the big files go
- Often contains a `raw` subfolder
  - Files that came from external sources, untouched
- Maybe a separate subfolder or subfolders for `processed` files
  - trimmed, filtered, concatenated, etc.
- Use sub-subdirectories for organization:
  - by processing stage
  - by date
  - by sample
  - *Consider*: it is often easiest to process ***all*** the things in a folder together; organize by units of work

Spend time thinking about this organization! It will reward you later.

The Naming of [Files] is a difficult matter

(apologies to T. S. Eliot)

Some “good” file names:

- `cheese-ratings.tsv`
- `2022-02-12_cheeseshop-inventory.tsv`
- `01_compile-ratings.py`

Some “not so good” file names:

- `script.py`
- `AVRS638GVEW4.fastq.gz`
- `tastingnotes (3).docx`
- `My favorite cheeses FINAL3 for Anatole UPDATED.docx`

EXTRA-'SPECIALLY GOOD Anatole
'SPECIALLY GOOD add a little vinegar Anatole
NOT SO GOOD needs more grated onion Anatole
NO GOOD Anatole
GOOD but add some more orange peel Anatole

image from *Anatole* by Eve Titus
illustrated by Paul Galdone

Jenny Bryan's principles for file naming (modified)

- machine friendly
- human friendly
- sortable and computable

Machine friendly

- Avoid spaces
    - Old computer systems get confused by spaces
    - All computer systems are old underneath
    - Use underscores or dashes to separate words instead
- Use “standard” characters:
    - Letters, numbers, underscores, and dashes
    - Periods only for file extensions (`.txt`, `.tsv`, `.R`, `.tar.gz`)
    - Many characters have special meanings in code. Avoid them! (e.g. `* + ? | $ / “`)
- Be consistent with case
    - Don’t *assume* case has meaning: on some systems it does, and on some it doesn’t
    - But always *act* as if it does!
        - Never have two files that are the same but for case

Human friendly

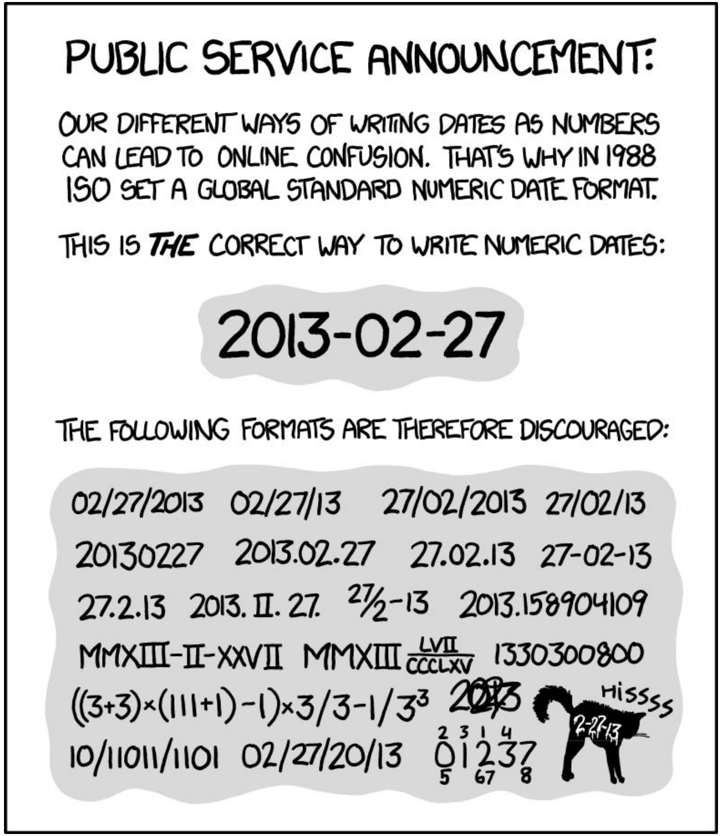
- Names should contain information about file content
- Short names are tempting, but you may regret choosing them!
    - `01.R`
    - `data.txt`
    - `tests.py`
- Use long descriptive names
    - `01_download-ena-data.sh`
    - `fig01_penguin-weight-histogram.png`

Which files do you want to look for before a deadline?

Which files do you want to get from your collaborator?

Sortable

- Use numbers for consistent sorting
  - `fig01_project-overview.pdf`
  - `fig02_sample-descriptor-histogram.pdf`
  - `fig03_oncoprint.md`
  - Left pad with `0` for consistent number length; this helps the computer sort properly
    - `7` is sad when it gets sorted after `11`
- Dates: use ISO 8601
  - Year-month-day is unambiguous and sorts nicely!
  - `2000-05-04_jedi-council-attendance.tsv`
  - `2000-05-05_sith-council-attendance.tsv`



PUBLIC SERVICE ANNOUNCEMENT:

OUR DIFFERENT WAYS OF WRITING DATES AS NUMBERS CAN LEAD TO ONLINE CONFUSION. THAT'S WHY IN 1988 ISO SET A GLOBAL STANDARD NUMERIC DATE FORMAT.

THIS IS *THE* CORRECT WAY TO WRITE NUMERIC DATES:

2013-02-27

THE FOLLOWING FORMATS ARE THEREFORE DISCOURAGED:

https://xkcd.com/1179/

Computable

- Use consistent name formats
    - Use file extensions
    - Separate “chunks” with underscores & keep consistent order

- “Wildcards” will be your friend:
    - **`*`** is the most common wildcard in UNIX
    - **`*.txt`**: refers to all files that end with `` `.txt` `` (hopefully all text files)
    - **`2020-01-*`**: all of the files from January 2020

Files you didn't create

- All the guidelines and suggestions for file names are great for files you create, but sometimes files come from other sources
  - If you are lucky, they will follow nice conventions! 🎉
  - but often they won't 😕
- To rename or not to rename, that is the question
  - Leaving the name as it was sent can make it easier to track in correspondence
  - Reasons to rename:
    - uniformative generic names: `data.txt`
    - add source or date information
    - converting spaces or other special characters (but try to write code that can handle these!)
  - *If you choose to rename, do it with a script and document the original name and source*