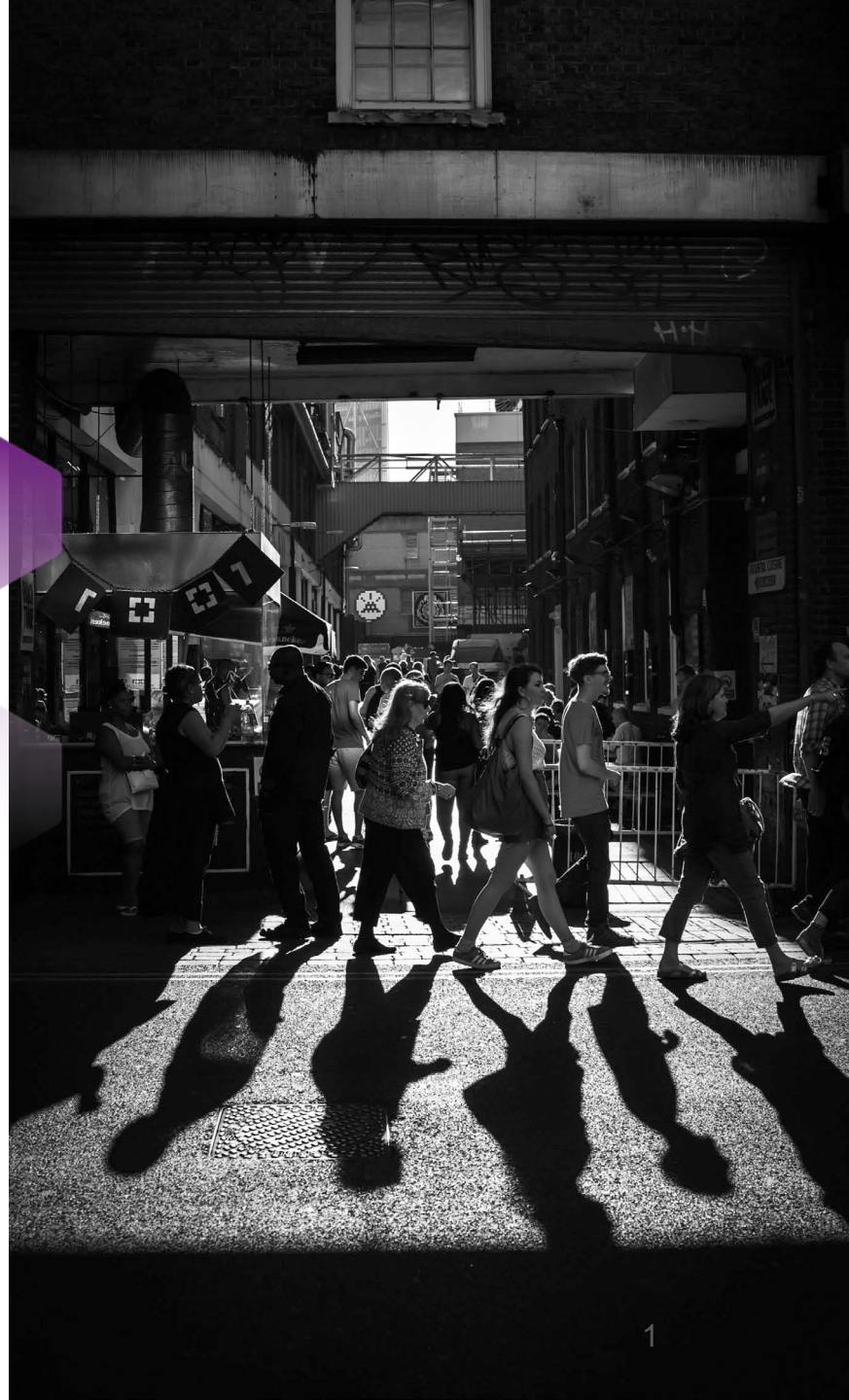
UK Data Service

Tools, data, code, and sharing

Coding in Public workshop

2024



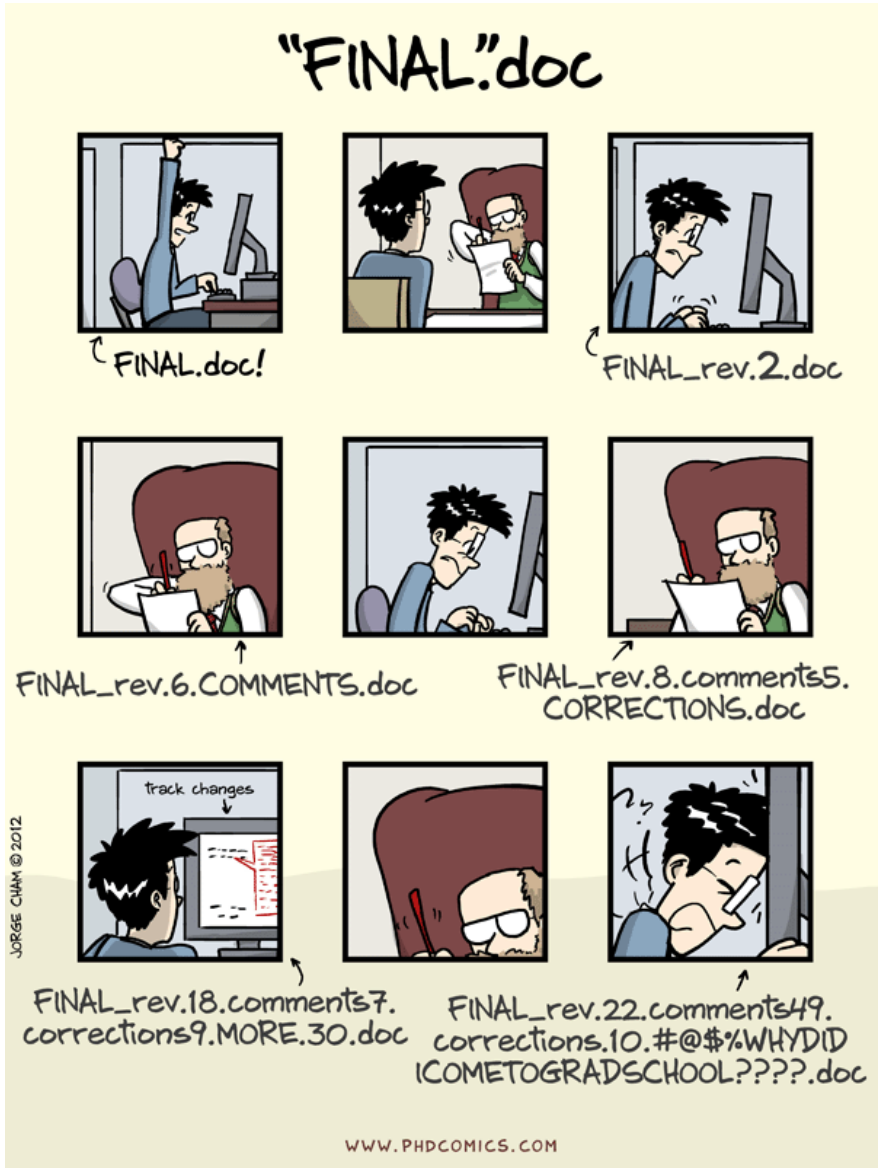
UKRI Economic and Social Research Council

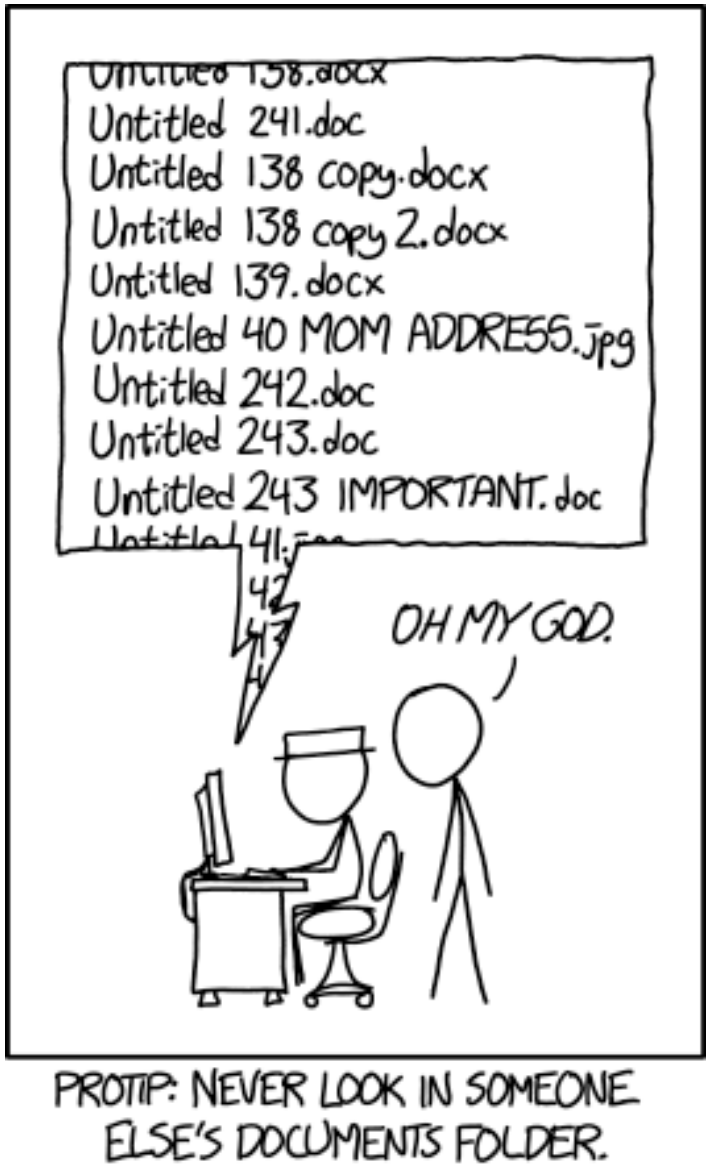
Table of Contents

UK Data Service

Modern tools

Tools can help prevent this sort of thing.





Tools to help

- Online meetings
  - Recording/auto transcription
  - Export attendees
  - Save and export chat
  - AI summarising/notetaking
- M365 suite – all in one solution?
  - Schedule meetings within a channel to save the chat
  - Send emails to channels
  - Save posts as to-do items
- Trigger-action services (IFTTT, Zapier, Pipedream, etc.)
  - Automate actions (export zoom attendees to a spreadsheet in gdrive, etc)
  - Useful if you need to work across lots of platforms
  - Requires access to accounts to run
- Version control – more on this next

Version control software

- Solves a lot of problems
  - Draft 1, draft 2, final draft 1, final draft 2, etc.
  - Integrating feedback from multiple users
  - Central storage
  - Change log

- Lots of options
  - In M365 (although relatively invisible so some users revert to old patterns)
  - Git and git-based cloud services
  - SVN, etc.
  - Very powerful, not so beginner friendly

Version control as a police evidence room

- The master version lives centrally
- People “check out” copies to work on
- The copies are checked back into the master version WITH
  - Changes, additions, etc.
  - A message about what was changes/added/etc., and
  - System stamps saying who checked it in and when
- Conflicts (incompatible changes on 2+ check ins) must be resolved
  - reject all check-in changes from one or the other
  - compare incompatible changes side by side and reject one at a time
  - etc.
- Good practice = always “check out” newest copy before working

Other topics

- Track tasks
  - To-do lists/apps
  - Kanban boards/apps
- Automation
  - Link apps/online services together to chain actions
  - Choose good triggers and actions based on agreed processes (e.g. when Kanban board ticket is marked as finished, add line to spreadsheet)

What tools do you like to use for teamwork?

Sharepoint

Notion (again)

Jira & Confluence really great for project management particularly agile projects. Also great for reporting on tasks backlogs. We use teams share point sites for documents

Teams , share point, Miro

Set up a slack channel for a group of colleagues, but it has yet to be used.

Modern pace of technological change

- Tools are made or discontinued at a rapid pace
- Software is updated frequently, complicating versions/dependencies
- Methods are developed continuously, but are not always very accessible
- Plus, cultish adherence/avoidance

Modern tools, modern problems

- You can automate the boring stuff to do more than was feasible before
- But automated processes can break easily

Ignorance can be fixed, but not everything can

- Rapid pace of change and increased complexity mean = lack of skills or knowledge
- Fixing this is possible, but not always easy
- Openness to change is harder because it is often a (nearly) fixed feature of personality

Share some of your wins and/or fails

FAILS: deleted our whole sharpoint site (many many years ago), we have documents still shared by email rather than one version & linked to.

Set up a slack channel for a group of colleagues, but it has yet to be used.
Probably my fault

Consider when choosing collaborative tools

- Cloud vs. local
- Cross-platform/open source vs. paid/proprietary
- Tracked vs. untracked
- Written vs. spoken
- Visual vs. descriptive
- Consider size/speed requirements
- Make environments clear (or irrelevant)

Tools for teamwork

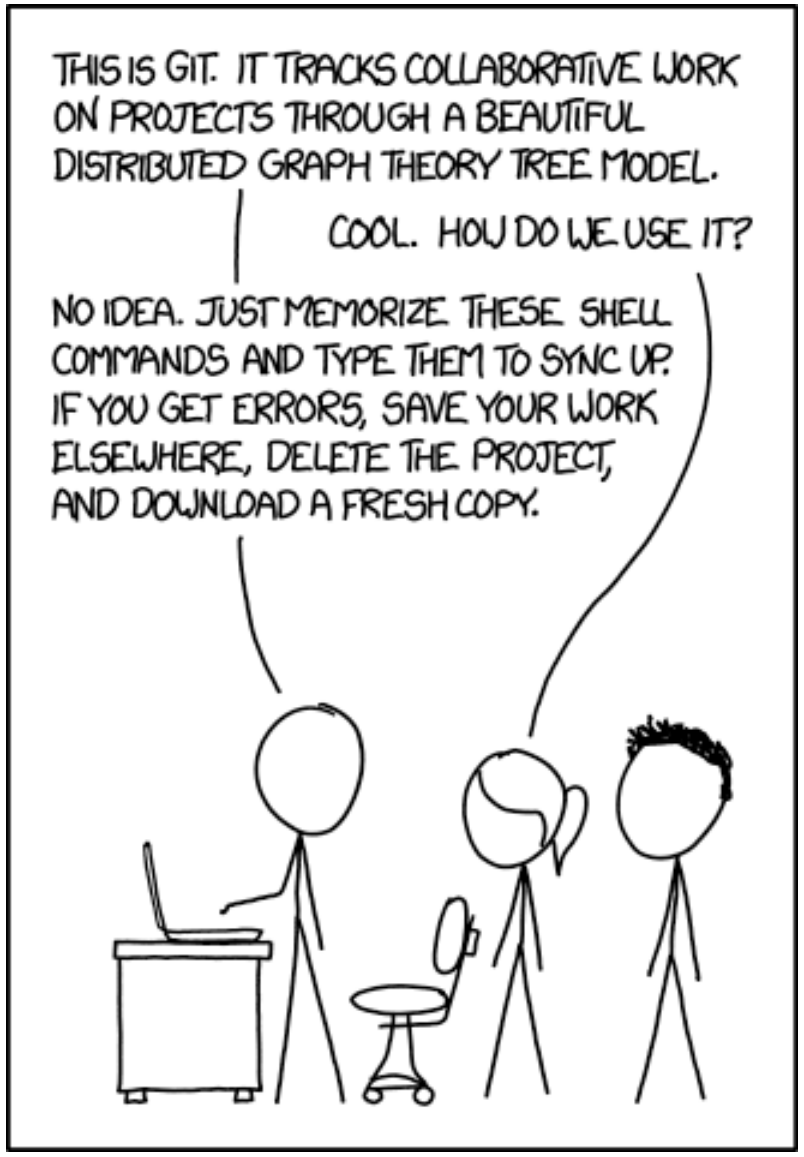
- Pick (and stick to) good naming conventions and shared locations
- ALWAYS leave useful commit comments (if available)
- Everyone uses most recent version each time (check out)
- Agree in advance (and stick to) a plan for settling conflicts

UK Data Service

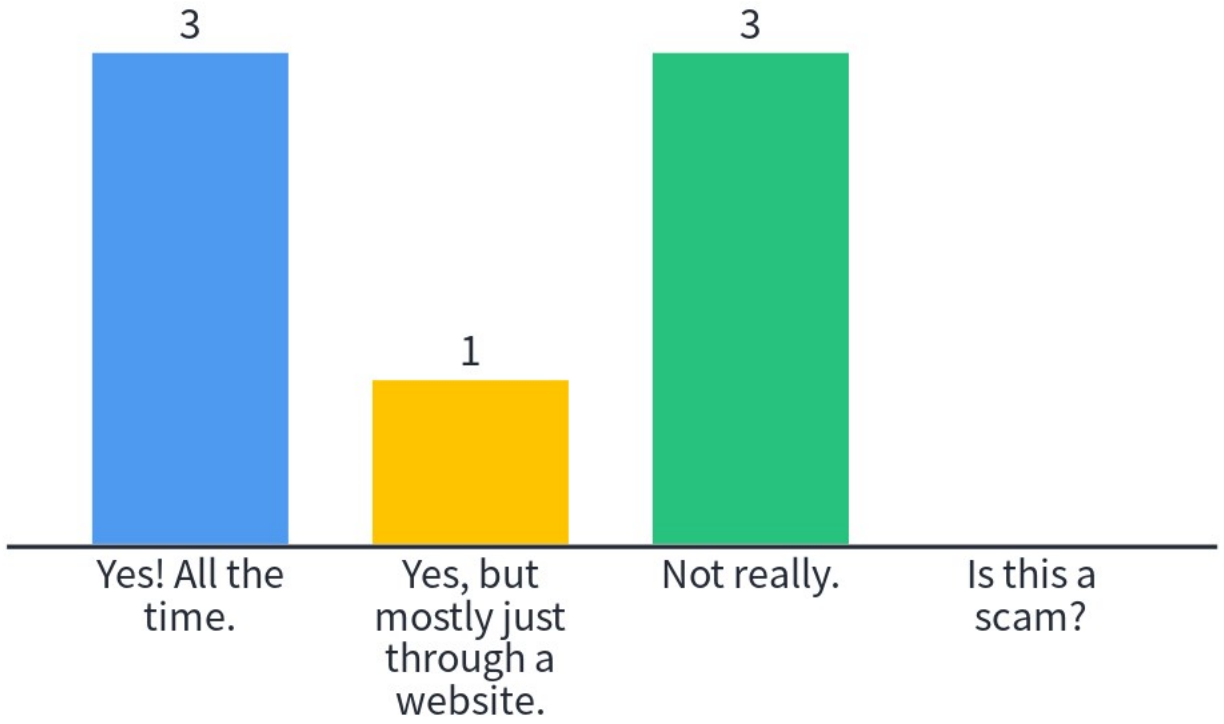
Command line and code

Shell commands and code can seem scary

But actually…



Have you used git?



Key shell commands for git

- git pull
- git add *
- git commit –m”useful commit message here”
- git push
- git status

What are your key git commands?

status, add, commit, push

pull, push, status, commit, merge (not often, and 9/10 times I regret it)

Git init, git branch, git checkout

Merge, -d

add, push, and commit

Collaboration, code, version control

Use “comments” in code to:

- Declare any useful background info (like author, current project status, software environment, etc.)
- Explain what this given line of code does or what it refers to
- Ask yourself questions, list steps you want to do next, address whether a functional thing will be upgraded in the future, etc.
- Example:

```
## Basic Coding Example by J. Kasmire
## Python Version- 3.9.5
print("spam and eggs")      ## This classic beginner command
                            ## references classic Monthy Python sketches
```

Code for what, exactly?

- All the steps of your process that you possible can do in code
  - Data collection
  - Cleaning/prepping
  - Analysis
  - Visualisations

But why code?

- By writing your processes in code, you
  - Are forced to write it correctly/ordered
  - Can rerun it/generalise it easily
  - Can share it
  - Have a head start or methodology section.

What barriers stop you and your colleagues from using code?

prefer to use github desktop. much simpler to use, don't have to write extra code

Everything you mentioned plus justifying the time to learn it, steep learning curve, and lack of support

Additional training needed, can be time consuming to teach the basics

Time to learn it. Also Different team members know different packages

Skills, confidence , IT. Lack of time to learn within projects.

Where/how should I code?

- Coding language appropriate for your project
- Cloud with password protect if needed (check GDPR)
- Interactive notebooks are ideal (more on this later)
- Capture your computational environment (more on this later)

Coding for teamwork

- Pick (and stick to) good naming conventions
- ALWAYS leave useful commit comments
- Everyone always checks out before working
- Agree in advance (and stick to) a plan for settling conflicts
- **Write in-code comments – LOTS OF THEM!**
- **Use good naming conventions for in-code variables too.**

UK Data Service

Documenting data

Data sharing – FAIR

- Findable – somewhere popular, indexed, well labelled, etc.
- Accessible – as open as possible
- Interoperable – in a standard format, shape, etc.
- Reusable – appropriate meta-data, column names, etc. so it is easy to combine, etc.

Data sharing – Supported locations

- Often provide DOI, permanent url or citations
- UKDS https://ukdataservice.ac.uk/deposit-data/
- UK Data Archive https://www.data-archive.ac.uk/deposit/
- OSF https://osf.io/
- Mendeley data https://data.mendeley.com/
- Field or discipline specific holdings https://www.bgs.ac.uk/geological-data/national-geoscience-data-centre/ngdc-depositing-data/
- University or research institute libraries?
- Your funding body?

Data sharing – Own locations

- More control, but also more responsibility
- Github
- Personal website
- Cloud server space
- Own server

- You can make your own DOI, permanent link or citation!
- QR codes make sharing easy www.qrcode-monkey.com

What are pros/cons of “data available on request”?

Cons - doesn’t mean you’ll get access

Con: it's a process to locate data owner, ask, wait, get the data and then realise it's not useful

pros: at least it can be available. better than no statement of data availability.

Most of the time "no response". old/ non used email ids provided

Makes it harder to reproduce. Unable to contact the person. Extra step.

If you can’t share your date, describe it! What is important in a good data description?

We try to include a data template &/or data dictionary. We are starting to consider synthetic but new to this.

Data collection method. Sample size. Variable names. Variable labels. Derived variable code. Creator and a ton more

dimensions, type of variables, size

Type, scope, collection methods, visualisations?

"Dataset contains information about... contains columns on x,y,z, with key variables"

Variables names with meta data, collection/sampling method, descirptive, any aggregation steps etc

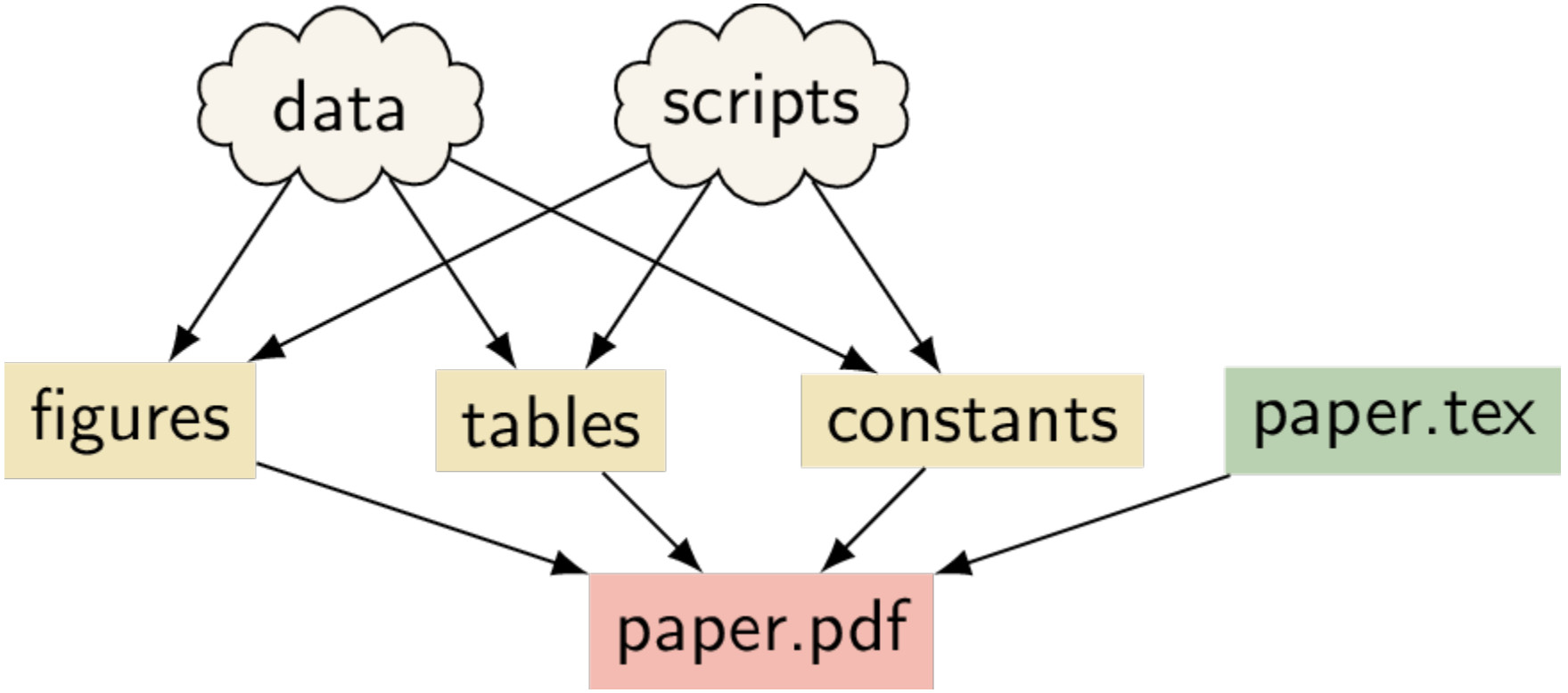
UK Data Service

Deployment and sharing

Capture your computational environment

What is reproduced? \ Interaction style	Graphical	Command line
Software & versions	binder	CONDA
Entire system		docker

Makefiles



Interactive notebooks

- jupyter notebook, Rmd, etc.
- Allow users to inspect/run code without downloading it
- The environment is built into the interactive notebook
- Can include comments, etc.

Sample jupyter notebook

Go to my arrow of time notebook?

References

- UKDS https://ukdataservice.ac.uk/deposit-data/
- UK Data Archive https://www.data-archive.ac.uk/deposit/
- OSF https://osf.io/
- Mendeley data https://data.mendeley.com/
- https://www.bgs.ac.uk/geological-data/national-geoscience-data-centre/ngdc-depositing-data/
- https://docs.github.com/en/repositories/archiving-a-github-repository/referencing-and-citing-content
- https://www.wpbeginner.com/wp-tutorials/how-to-create-custom-permalinks-in-wordpress/
- www.qrcode-monkey.com
- Data in Government Blog tinyurl.com/Synth-DataInGov
- The unreasonable effectiveness of synthetic data with Daeil Kim tinyurl.com/Synth-Podcast
- Medium article this webinar is based on tinyurl.com/Synth-Blogpost
- The Anonymisation Decision-Making Framework https://msrbcel.files.wordpress.com/2020/11/adf-final-version-3.1-for-uploading.pdf
- Synthetic data estimation for the UK longitudinal studies https://calls.ac.uk/output-entry/synthetic-data-estimation-for-the-uk-longitudinal-studies-sylls-project-an-introduction-to-the-multiple-imputation-approach/
- Synthetic Datasets for Statistical Disclosure Control: Theory and Implementation: 201 https://www.springer.com/gp/book/9781461403258
- https://www.nature.com/articles/s41551-021-00751-8
- Make https://the-turing-way.netlify.app/reproducible-research/make

UK Data Service

Q&A

UK Data Service

Breakout #3