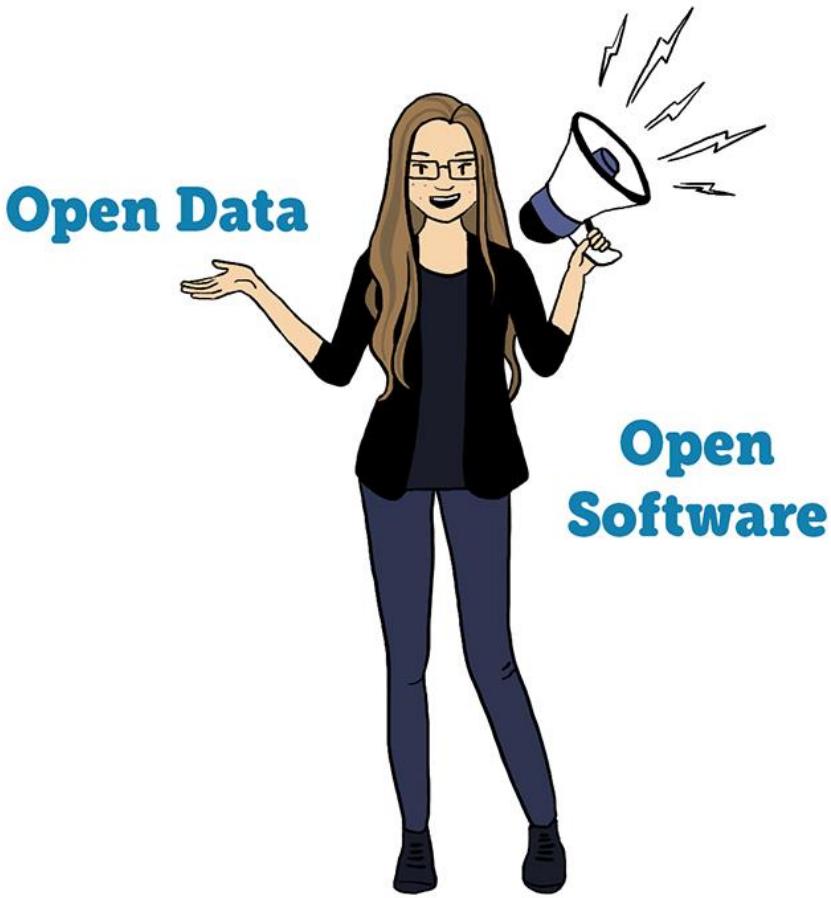
Open Science: Crash Course

Open Data/Software: What, when, how?

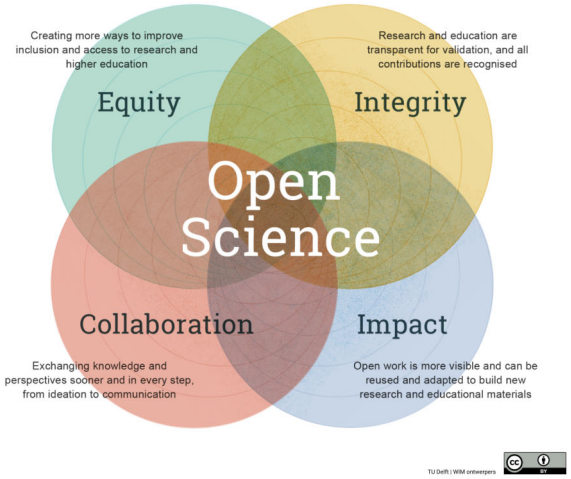
Slides by **Esther Plomp** @ TU Delft, Faculty of Applied Sciences

Link slides



Open Science

Open Science



Open Data

Open Software

Open Education

Open Hardware

Open Methods

Open Publishing

Collaboration (community, inclusion)

Citizen Science

tudelft.nl/openscience

Open Data

Planning for Open Data

Data Management Plan (DMP) to plan how to manage and share the data (see The Turing Way for more information)

TU Delft has access to DMPonline with TU Delft specific templates and guidance

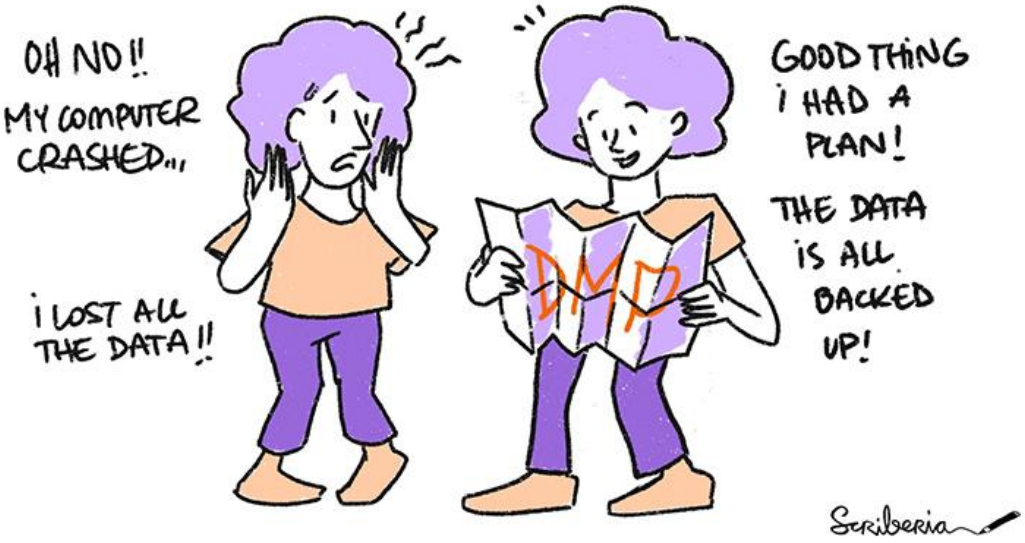


image by The Turing Way

Data Organisation

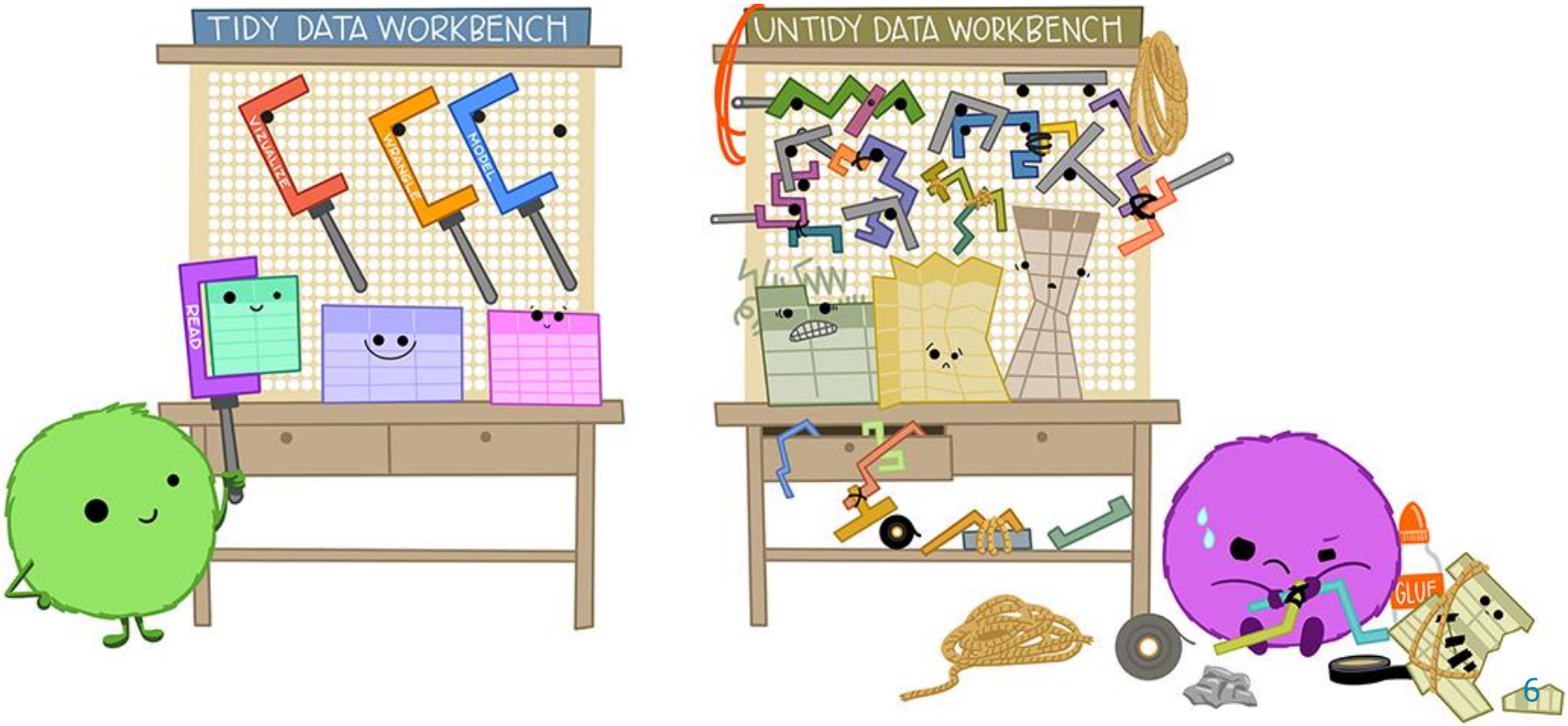


Image by Allison Horst

File naming

- 20220113-PRES-Data-V001
- 8 step guide on how to set up your file naming convention
- Presentation on file naming
- Stanford's best practices

Folder structure

- Templates by Colomb et al., Nikola and Barbara Vreede for cookiecutter
- Find Files Faster: How to Organize Files and Folders
- Data Management: File organisation by Christine Malinowski
- Videos on project structure by Danielle Navarro
- Software: Cookiecutter template by Barbara Vreede based on Wilson 2017

Data Organisation

Spreadsheets

- Spreadsheet organisation tips
- Broman and Woo 2018
- Wickham 2014
- Use tools for data validation like OpenRefine

Why? What could possibly go wrong?

- a lot

Data Documentation

- (electronic) Labnotes: TU Delft provides licenses for eLABjournal and Rspace
- Readme files (template)
- Guide for data documentation
- Data Dictionary
- Code Book

More information

- Book: Data Management for Researchers by Kristin Briney
- A Quick Guide to Organizing Computational Biology Projects by William Noble
- Some Simple Guidelines for Effective Data Management by Borer et al.

Open Data

made freely available for use and re-use by anyone and everyone

Access : Available (on the internet) to all on demand

Reuse/distribution : Provided under terms that permit reuse and redistribution

Transparency : Providing information about data generation/collection

Interoperability : Interoperability with other data, machine readable formats

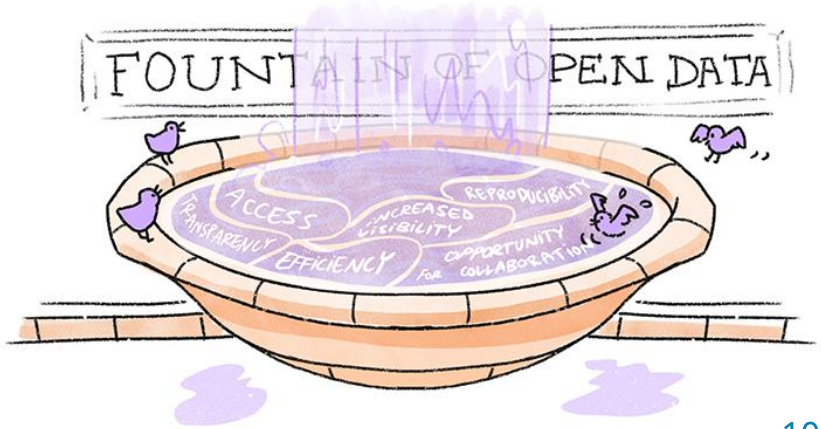
Participation : Everyone must be able to use, reuse and redistribute

Equity : Data is not truly open if the research process is not open to all

#bropenscience is broken science by Kirstie Whitaker and Olivia Guest

Open Science Beyond Open Access: For and with communities

image by The Turing Way



Not Open Data

'[odds of obtaining the dataset] fell by 17% per year' Vines et al. 2014

‘research data cannot be reliably preserved by individual researchers’ - Vines et al. 2014

"We received no response to 41.3% of our data requests" - Tedersoo et al. 2021

Meme explanation

Open Data

data repository

online archive that curates research datasets and provides long-term access

- Finalised datasets
- ~10-15 years (Long term preservation)
- Access
- DOI = more citations/visibility
- File format support

How can you make research data accessible? by Esther Plomp

Open Data

How to find a repository?

- Check publications in your field
- FAIRsharing
- re3data

General repositories:

- 4TU.ResearchData
- Zenodo

Open Data

A **Data Article** (also known as a Data Paper/Note/Release, or Database article) is a publication that is focused on the description of a dataset.

More information on The Turing Way and TU Delft specific information

Open Software

Open Software

Software in which the copyright holder has granted a license to **use, study, change, and/or distribute the source code.** - opensource.org

Sharing Software allows for

- Scrutiny of methods / increased **reproducibility**
  - see Krafczyk et al. 2021 p5-11 for recommendations
- **Collaboration**
- **Credit**

See also: Making software FAIR? for more resources

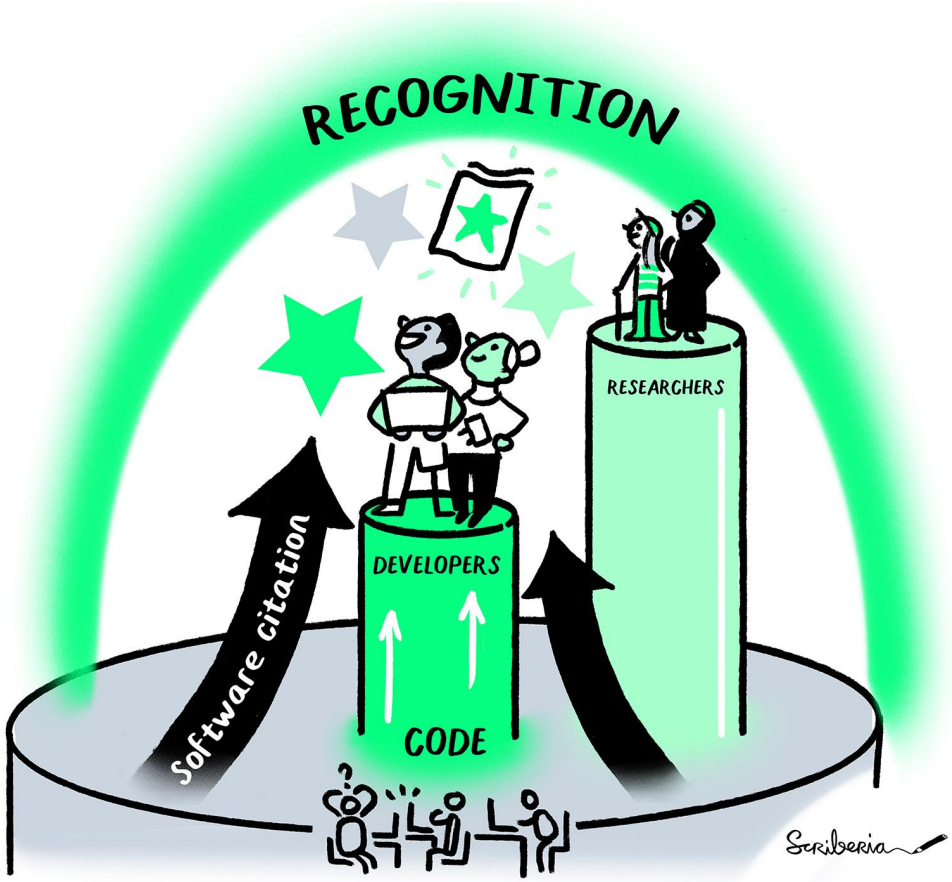


image by The Turing Way

Version control

Version control allows you to **easily track changes**, both your own changes as well as those made by collaborators (for example, Git)

By configuring your version control system to use GitHub, GitLab or Bitbucket, you'll have **backups** of every version. Follow this webinar for an introduction to GitHub. You can also use TU Delft GitLab.

These platforms offer **collaboration tools** (issue tracker and project management tools), and you'll be able to use third-party **services** such as code quality checkers, correctness checkers.

image by The Turing Way]

Also useful if you do not code

- Working together on projects (Open Life Science, The Turing Way)
  - Live demo of collaborative working without code on GitHub
- Setting up your website (see Esther's website)
- Making your work available to others (slides, newsletters)
- Keeping track of other projects (stars)
- Project management tools (Project Boards, Issues)

README

- Landing page for your repo (watch this video for more info)
- The 'Abstract' of your project
- Who is involved in the project?
- Invitation to others to contribute (what expertise is needed?)
- On GitHub this is rendered in Markdown (language to format text)
  - Markdown Cheatsheet
  - Emoji cheatsheet

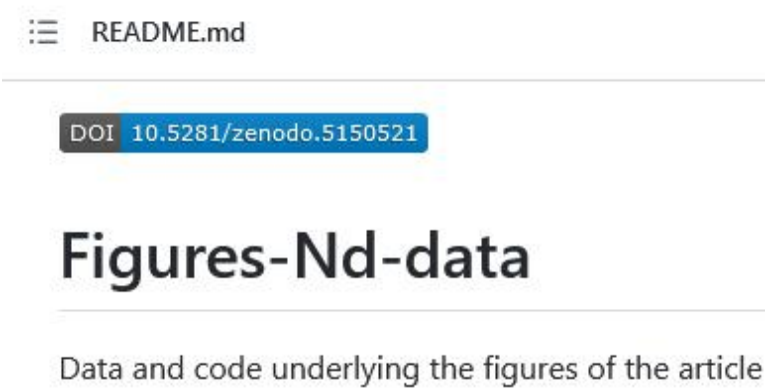
Examples:

- Jupyter & scikit-learn (written in Python)
- Matpower (written in MATLAB)
- Examples and explanation by Alex Chan
- Figures underlying Esther's article
- Templates: #1,#2

Getting a DOI for your GitHub Repository

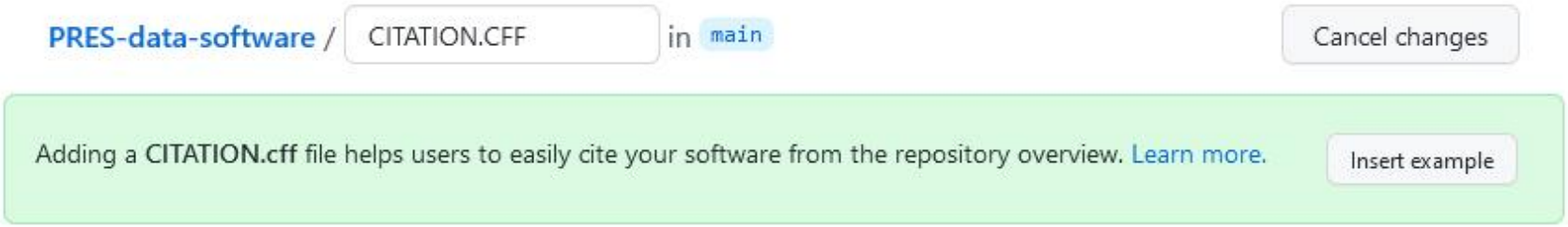
Summary of this guide

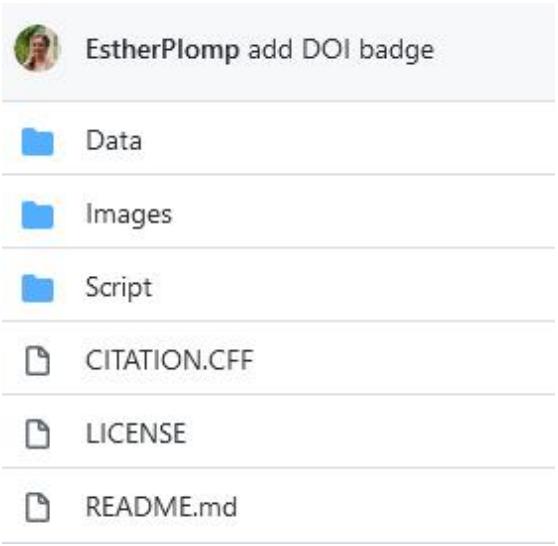
1. Have a repository in mind that you're creating the identifier for
2. Set up an account at Zenodo (using your GitHub, email or ORCID)
3. Connect GitHub/Zenodo by authorizing Zenodo
4. On Zenodo, go to Settings -> GitHub, then toggle the button on for the repository that you want an identifier for
5. On GitHub: create a new release for your repository (snapshot that will be preserved on Zenodo)
6. On Zenodo: go to the Upload tab and add any additional information before publishing.
7. On GitHub you can update your citation file with the DOI and add a DOI button in your Readme file



Software Citation

- Create a new file in your repository, name it CITATION.cff, select insert example, and fill out the template:





More information: The Turing Way / video (& slides) with tips

Options for software sharing/publishing

	Code repository	Deposit in digital repository	Produce runnable version	Register in catalogue / registry	Paper in software journal	Paper in domain-specific journal
Example	Source code is in GitHub, GitLab or BitBucket with open license	Source code deposited in Zenodo, Figshare or an institutional repository	Jupyter Notebook in Binder, Capsule in CodeOcean, Docker or Singularity container, NextFlow workflow. Package for CRAN, PyPI, etc	Create an entry in a community registries e.g. ASCL (astronomy), CIG (geodynamics), RRID, swMath (mathematics). NLeSC RSD.	Publish software paper in JORS, JOSS, SoftwareX, etc. Publish executable research article in GigaByte	Many journals now accept papers about software – see bit.ly/softwarejournals
Advantages	Discoverable Fits with development workflow No waiting before available	Archived Persistent identifier and metadata Little/no wait before available	Enable direct reuse Can be given identifiers Makes available in location where users search	Indexed Easier to find Often provides identifier May show citations	Easily citable Peer reviewed Can describe software design Easier for developers to write	Easily citable Easier to reach target audience Understood by promotion committees
Disadvantages	Not archived Harder to cite Not easy to find if poorly described / documented	Direct software citations not accepted by all journals	Normally requires additional effort / resources	Not available in every domain Many people just Google, so must be indexed	Software not always archived Not as "prestigious" as domain-specific journal	Software generally not archived. Longer time to publishing. Not easy to run.

Checklist for software sharing

- Have I assigned an **appropriate license** to my software?
- Have I **described my software properly**, using an appropriate metadata format, and included this metadata file with my software?
  - Have I given my software a clear **version number**?
  - Have I determined the **authors to be credited** for this release of my software, and included this in my metadata file?
- Have I procured a **persistent identifier** for this release of my software?
- Have I added my **recommended citation** to the documentation for my software?

Checklist for developers: https://doi.org/10.5281/zenodo.3482769

If you're reusing Software of others

- Have I **identified the software** which makes a significant and specialised contribution to my academic work?
- Have I checked if the software has a **recommended citation**?
  - If this is to a paper, have I also cited the software directly?
  - If there's no recommended citation, have I **created as complete a citation as possible**?
    - Who created the software
    - When it was created
    - Title of the software (and version if available)
    - Where the software can be accessed
- Have I **referenced the software appropriately** in my academic work, complying with any citation formatting guidelines?

Checklist for authors: https://doi.org/10.5281/zenodo.3479199

Why not supplemental materials?

Why not supplemental materials?

Data control: cannot be updated

Interoperability: not available in all formats which makes it difficult to integrate and interact with the data

Availability: Difficult to access if the article is behind the paywall (supplemental materials are not included in the DOI and therefore the links can also break!)

Impact: Data should be a primary research output

Publisher requirements: Some publishers recommend using a data repository instead

Not FAIR: Data/Software available in supplemental materials is not considered to be FAIR (Findable, Accessible, Interoperable, Resuable)

See also: The Push to Replace Journal Supplements with Repositories

Sustainable access to data/code

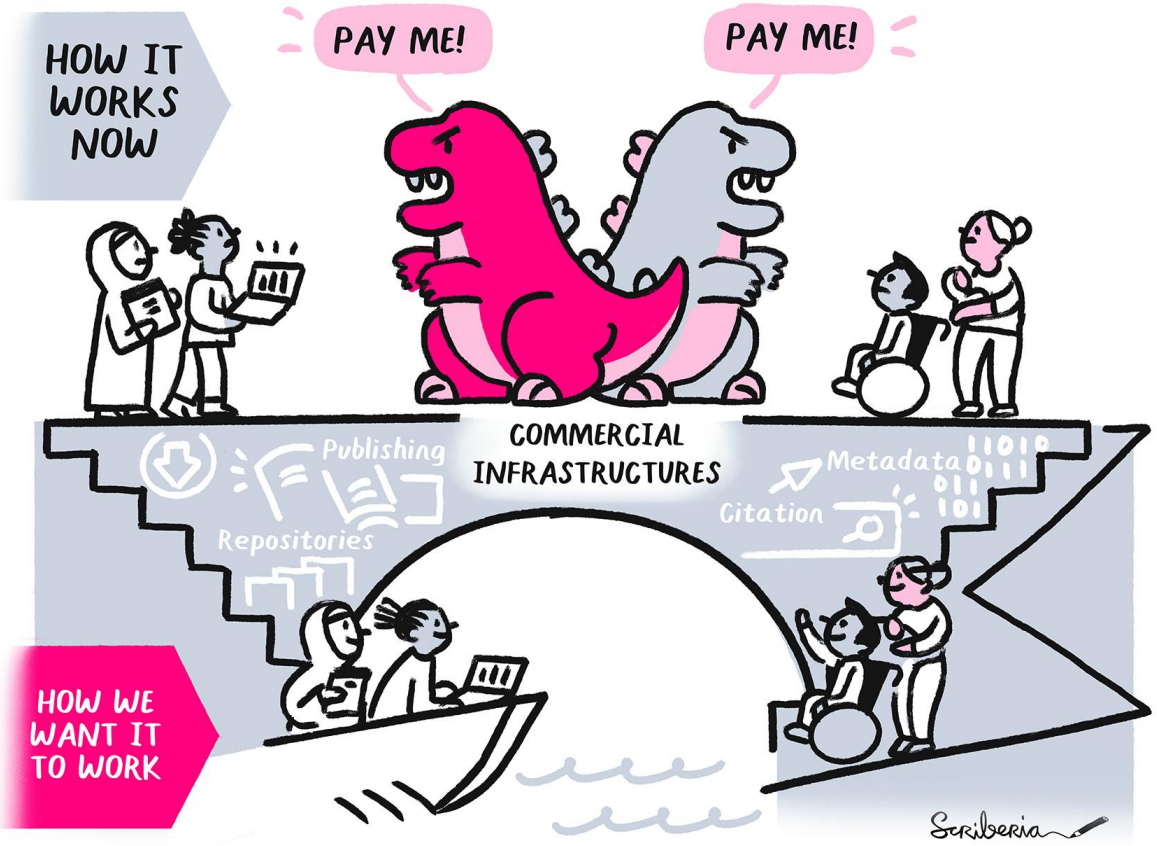
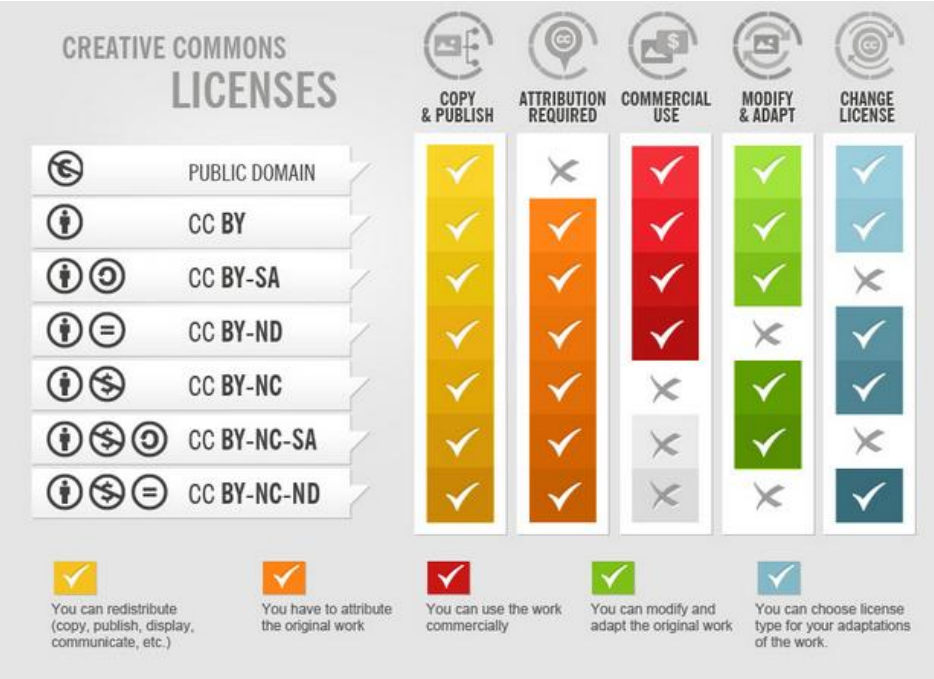


image by The Turing Way

Licenses

Licenses



Data

Creative Commons License Chooser

Software

Choose an open source license

Video on licenses

Image Source: CC-BY-SA

Software Licenses

Permissive

Open licenses that do not require derivative works to shared with the same license.

Examples:

- CC BY
- MIT, BSD, APL-2.0

TU Delft approved licenses according to the TU Delft Research Software Policy and Guidelines

Copyleft

Open licenses that require all derivative works to be shared with the same license.

Examples:

- CC BY-SA
- GPLv3, AGPL, LGPL, EUP

Sharing software according to TU Delft

Choose a pre-approved license (MIT, BSD, Apache, GPL, AGPL, LGPL, EUPL, CC0)

Use 4TU.ResearchData

- Log in (top right) using TU Delft credentials
- Create a new item or import from GitHub/GitLab
- Add relevant metadata in the information fields

OR

Use another repository (Zenodo) AND register the software in PURE

- Log in using TU Delft credentials
- Select Datasets/Software -> Software
- Fill out the metadata in the information fields and add DOI, select license and save the information

Full slidedeck + recording of how to publish software (from 23:14 onwards)

How to link your publication and data/code?

How to link your publication and data/code?

- Publish the output before you publish the article

OR

- Reserve the DOI

Use the DOI/citation in your publication

Reference your data in the **Data Availability Statement** and the **References**

Publish or reserving a DOI

Zenodo -> Upload -> New Upload

Linking with publication

Data accessibility/within table (descriptions)

Data accessibility	Repository: IsoArcH [1] Data identification number: 10.48530/isoarch.2021.011 Direct URL: 10.48530/isoarch.2021.011 Software availability: https://doi.org/10.5281/ZENODO.5150520 [6] Data is available under the Creative Commons BY-NC-SA 4.0 license.

Data availability statements (at the end)

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available in Tables 2-4 as well as openly available at the 4TU.Centre for Research Data (Plomp, Verdegaal-Warmerdam, & Davies, 2020, http://doi.org/10.4121/uuid:f6dc4f20-a6e0-4b2f-b2f8-b79a4f9061c3).

Linking with publication

References

[1] K. Salesse, R. Fernandes, X. de Rochefort, J. Brůžek, D. Castex, É. Dufour, IsoArcH.eu: an open-access and collaborative isotope database for bioarchaeological samples from the Graeco-Roman world and its margins, J. Archaeol. Sci. Rep. 19 (2018) 1050–1055, doi:10.1016/j.jasrep.2017.07.030.

[5] E. Plomp, Neodymium isotopes in modern human dental enamel: an exploratory dataset, IsoArcH (2021), doi:10.48530/ISOARCH.2021.011.

[6] E. Plomp, J.C. Peterson, [software] EstherPlomp/figures-Nd-data, Zenodo, 2021. doi:10.5281/ZENODO.5150520.

[7] C. Stantis, [software] stantis/IsoDataVis: first (Official) release, Zenodo, 2021. doi:10.5281/ZENODO.4743734.

Always check the dataset's readme file or metadata on how the contributors prefer to be cited!

See this document for more information about data/software citation.

Linking data/code/publication

Nanopore electro-osmotic trap for the label-free study of single proteins and their conformations

Sonja Schmid[1,3], Pierre Stömmer[2], Hendrik Dietz[2] and Cees Dekker[1]

https://doi.org/10.1038/s41565-021-00958-5

Data availability
Data are available at https://doi.org/10.5281/zenodo.5059802.

Code availability
Code for data analysis of nanopore recordings as described herein are available at https://doi.org/10.5281/zenodo.5059802.

July 2, 2021

Dataset Open Access

NEOtrap data

Sonja Schmid

Source data and code used in "Nanopore electro-osmotic trap for the label-free study of single proteins and their conformations" by Schmid, Stömmer, Dietz, Dekker (2021) Nature Nanotechnology.

Publication // Data & Code

Where next?

Where next?

TU Delft Open Science Community

> Sign up for a 2-monthly newsletter, Slack channel and visibility on the website.

Open Life Science Programme

> PhD candidates can follow this programme for 5 disciplinary specific credits. See intranet for more information. New applications will open over the summer with the programme taking place around September-December.

The Turing Way

> There will be an online/hybrid event to contribute to the Turing Way in May/June. Contact Esther for more information.

Thanks!

Slides created via the R packages:

xaringan
gadenbuie/xaringanthemer

remark.js, **knitr**, and R Markdown

images by The Turing Way