Research data management and publishing at your fingertips!

fairly Toolset

National Open Science Festival, 31/08/2023
Erasmus University Rotterdam

Serkan Girgin

UNIVERSITY OF TWENTE. | ITC

Manuel Garcia Alvarez

TU Delft

Jose Urra Llanusa

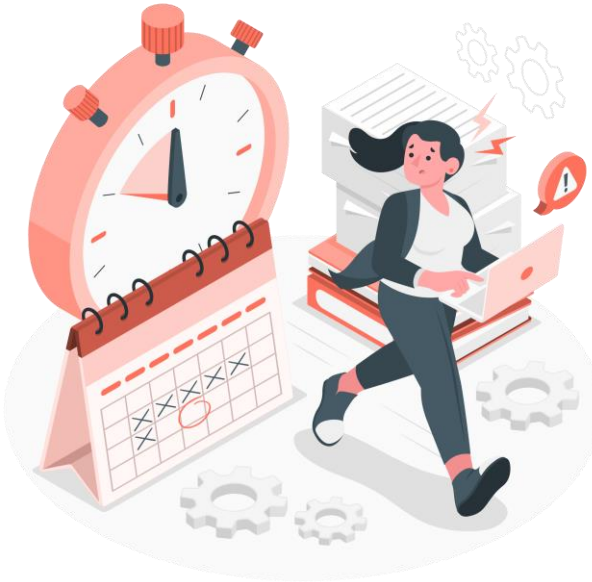
TU Delft

Magno Barreto de Araujo

TU Delft

We usually publish research data at **the last minute**

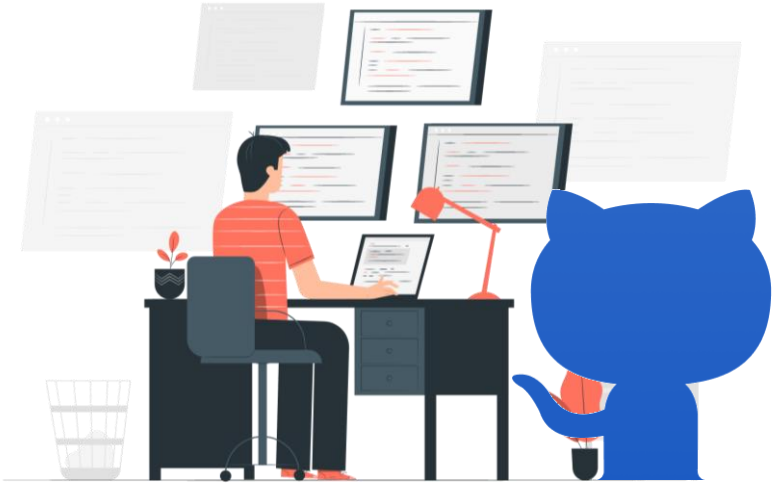
- Research data are produced during the whole research lifecycle.
- Data publication and sharing happens mostly at the end.
- Data published in a hurry lack important supplementary information and metadata limiting reusability.
- On the other hand, periodic high-quality data publication takes time and effort.



https://storyset.com

We cannot publish research data directly from our **digital research environments**

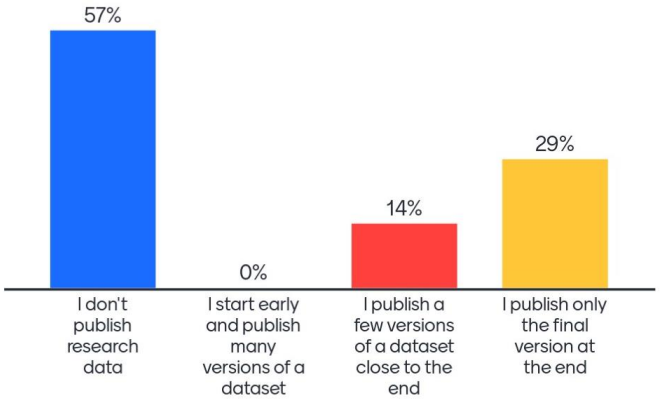
- Digital research environments facilitate research data production by providing (interactive) analysis tools.
- They are well connected to some research infrastructure, e.g. code repositories.
- However, their interoperability with research data repositories is weak.



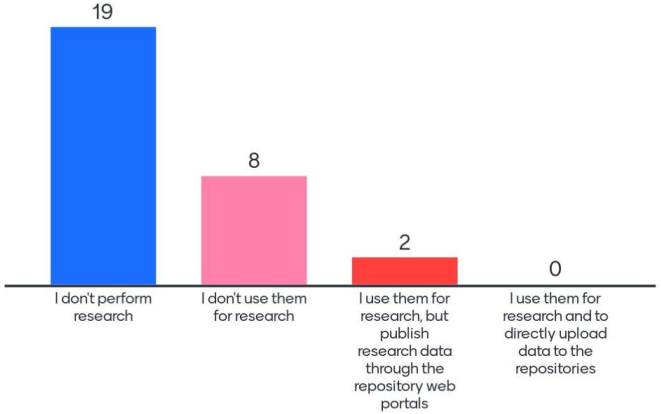
https://storyset.com

What do **you** think?

When do you publish your research data?



How do you use digital research environments?



Seamless integration of research environments and data repositories may facilitate data management practices

How to integrate?

- Local datasets with data and metadata
- Direct and simple data transfer
- Less data input through forms
- Onsite quality checks

What are the benefits?

- Less time and effort for research data publishing
- More frequent data sharing during research lifecycle
- Improved quality of shared research data

https://storyset.com

Our project aimed at enabling **local management** and **easy publishing** of research data

- Design of a methodology to integrate research environments to research data repositories
- Development of a modular open-source software tool implementing the methodology
- Demonstration at 4TU.ResearchData and ITC Geospatial Computing Platform
- Provision of technical documentation and end-user training

Funded by the NWO Open Science Fund, File No. 203.001.114

For more information, please check the project proposal:

DOI 10.5281/zenodo.6026285

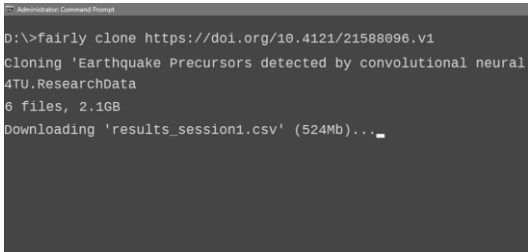
UNIVERSITY OF TWENTE. | ITC

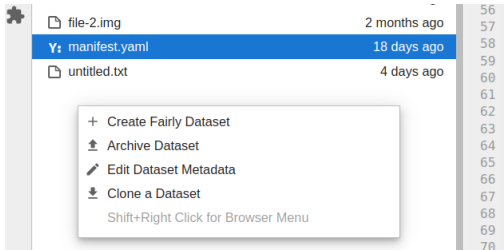
TU Delft

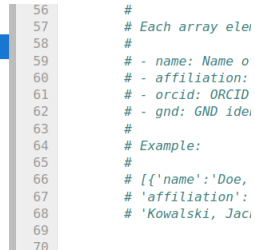
4TU.ResearchData
SCIENCE • ENGINEERING • DESIGN

We designed a **three-tier architecture** to serve different needs

1. Python package: **fairly**
   - Provides an API to create and manage research data by using Python
   - Enables further development by interested parties
2. Command line interface: **fairly CLI**
   - Provides commands to create and manage research data
   - Enables RDM without programming
3. JupyterLab extension: **jupyter-fairly**
   - Enables RDM inside a virtual research environment

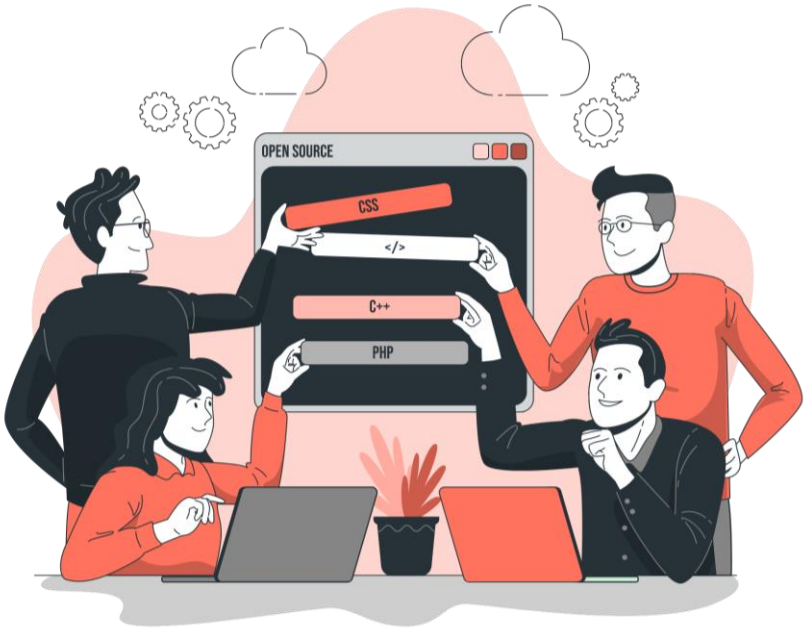






Open-source software developed by following best practices

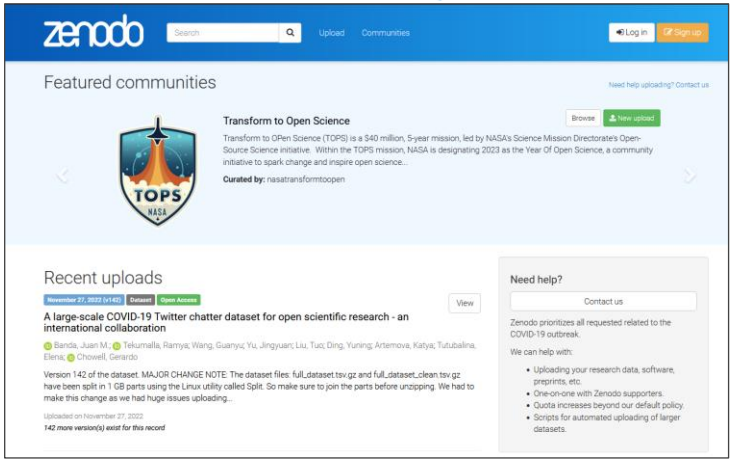
- Open-source Python code
  - Continuous integration (Github) and unit testing (pytest)
  - Documented source code (Google style guidelines)
- Object-oriented modules
  - Task-oriented classes for different components of RDM
  - Easily extendable by implementing abstract classes
- Minimum dependency on 3rd party packages
  - Direct use of repository platform REST APIs



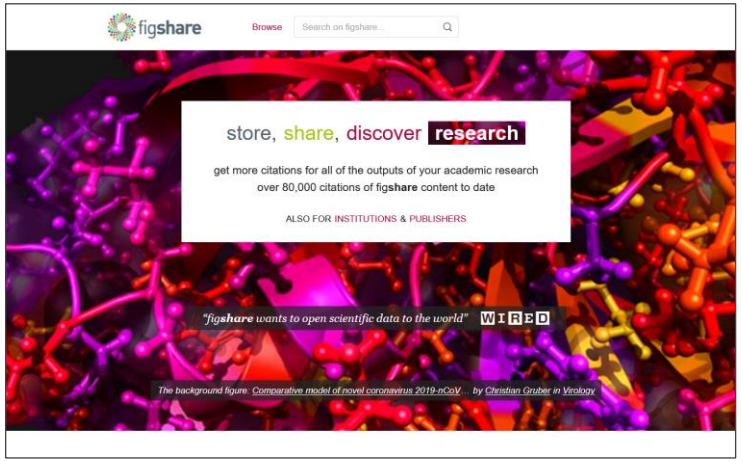
https://storyset.com

We implemented support for **multiple repository platforms**

INVENIO



figshare

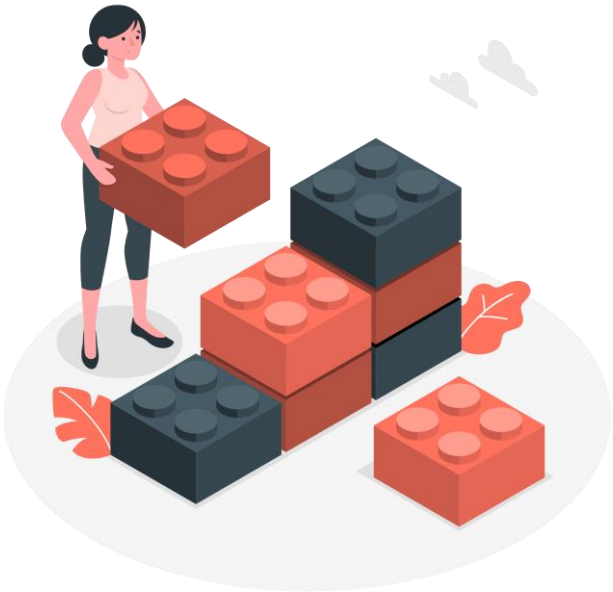


(more is coming soon!)

Dataverse®

A rich set of features is available for **efficient** data management

- Quick research dataset cloning
  - One-command retrieval of metadata and all data files by using URL address, DOI, or record identifier
  - Automatic extraction of archived data files (e.g. .zip, .tar.gz)
- Local metadata management
  - Creation and editing of metadata locally by using your favorite text editor or API methods
- Quick dataset publication
  - One-command creation of research data records at online data repositories in a unified way



https://storyset.com

A rich set of features is available for **smart** data management

- Unattended large dataset uploading
  - Easy uploading of a high number of data files and folders, including large files
  - Automatic creation of archive files (e.g. .zip, .tar.gz) if folders are not supported by the data repository
- Smart dataset synchronization
  - Automatic identification of added, removed, or modified files and folders
  - Upload / download of files and folders only if necessary
  - Easy versioning of datasets in a unified way considering the repository rules COMING SOON

https://storyset.com

How to access a **remote dataset** and **store** it locally?

```python
import fairly

# Open a remote dataset
dataset = fairly.dataset("doi:10.4121/21588096.v1")

# Get dataset information
dataset.id
{'id': '21588096', 'version': '1'}

dataset.url
'https://data.4tu.nl/articles/dataset/.../21588096/1'

dataset.size
33339

len(dataset.files)
6

dataset.metadata
Metadata({'keywords': ['Earthquakes', 'precursor'], ...
'online_date': '2022-11-24T07:50:39'})

# Update metadata
dataset.metadata["keywords"] = ["Landslides", "precursor"]
dataset.save()

# Store dataset to a local folder (i.e. clone dataset)
local_dataset = dataset.store("/path/dataset")
```

Import **fairly** package

Open dataset by using URL, DOI, or id
(creates a lazy remote dataset object)

Access dataset information
(retrieves information and caches it if possible)

Modify metadata attribute(s)

Save changes
(updates dataset record on the repository)

Store dataset as a local dataset, i.e. clone locally
(downloads and stores metadata + data files)

How to create a **local dataset** and **deposit** it to a repository?

```python
import fairly

# Initialize a local dataset
dataset = fairly.init_dataset('/path/dataset')

# Set metadata
dataset.set_metadata(
    title="My dataset",
    keywords=["FAIR", "data"],
    authors=[
        "0000-0002-0156-185X",
        {"name": "John", "surname": "Doe"}
    ]
)
dataset.metadata["license"] = "CC-BY-4.0"

# Add data files and folders
dataset.includes.extend([
    "README.txt",
    "*.csv",
    "train/*.jpg",
])

# Save dataset
dataset.save()

# Upload to the data repository
remote_dataset = dataset.upload("4tu")
```

Import **fairly** package

Initialize a local dataset at the specified folder
(creates a manifest file with the default metadata template)

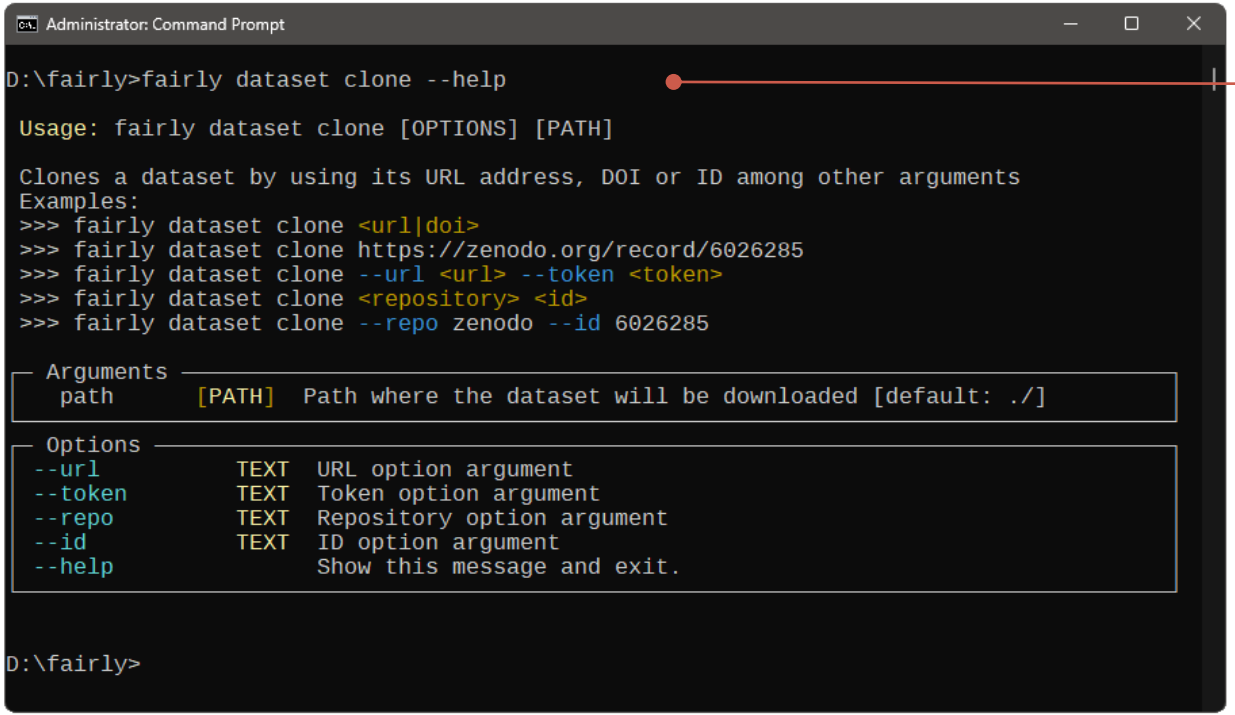
Set metadata attributes
(attributes can be set together or individually)

Set data files
(files to be included or excluded can be indicated by name or pattern)

Set dataset manifest
(manifest file is updated)

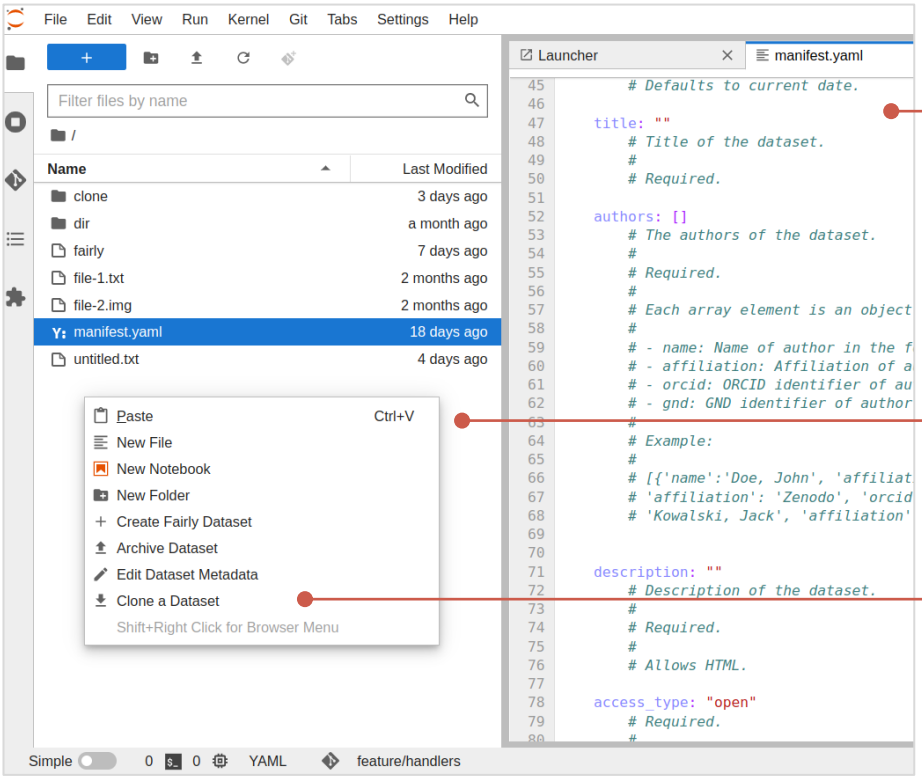
Update dataset to a data repository, i.e. deposit
(creates dataset record, sets metadata, and uploads files and folders)

How to access a **remote dataset** and **store** it locally *easily*?



Run **fairly** command line interface

How to access a **remote dataset** and **store** it locally *more easily*?



Enable **jupyter-fairly** extension

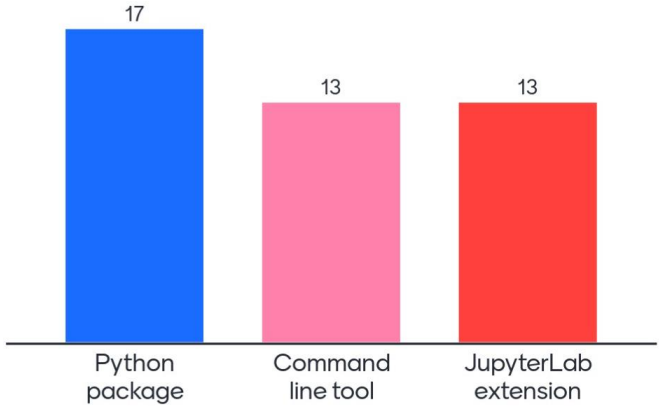
Open context menu
(right click)

Select 'Clone a Dataset'

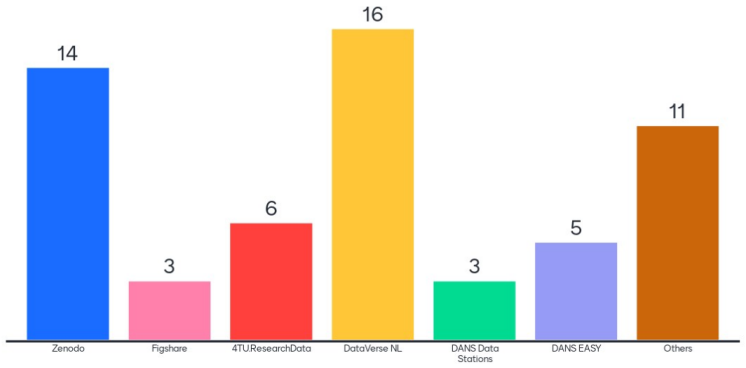
Enter URL address or DOI, click 'Ok'

Do they sound **interesting**?

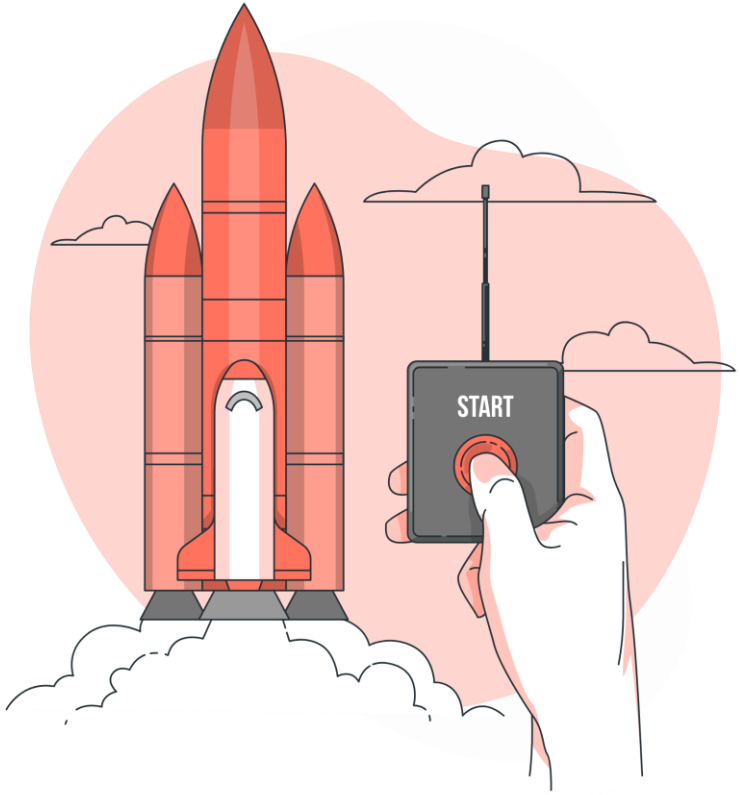
Which fairly tools might be useful for you?



Which research data repositories do you use?



Let's **try** together!



https://storyset.com

Let's **discuss** together!

https://storyset.com

Can you think of some potential **use cases**?

- To have a **standard** research data development workflow, like the use of `git` for research code?

  Can help changing research data management culture.

- To deposit **large** datasets?

  Can help uploading complex and big datasets.

- To publish updated versions of datasets **periodically**?

  Can help automatizing update processes.

- To **embed** research data management into workflows?

  Can help developing improved research workflows.

https://storyset.com

We are developing a platform to provide **analysis-ready exploratory research environment** with data

- Development of an **open-source software** to create and manage interactive computing environments with analysis-ready data
- Development of **template interactive notebooks** to facilitate rapid exploratory data analysis
- Operationalizing of a **prototype platform – opendataexplorer.org**
- **Feasibility and benchmarking study** to use the SURF infrastructure
- Development of the user **documentation and training material**
- Organizing a **training workshop**

Funded by the SURF DCC Investment for Digital Infrastructure Call

Illustration by Storyset.com

Join us to develop the community in a **sustainable** way

- **Co-design**

  Voice your ideas to improve the methodology according to the needs of different research disciplines and communities.

- **Testing**

  Test the tools and provide feedback to correct issues and improve features.

- **Co-development**

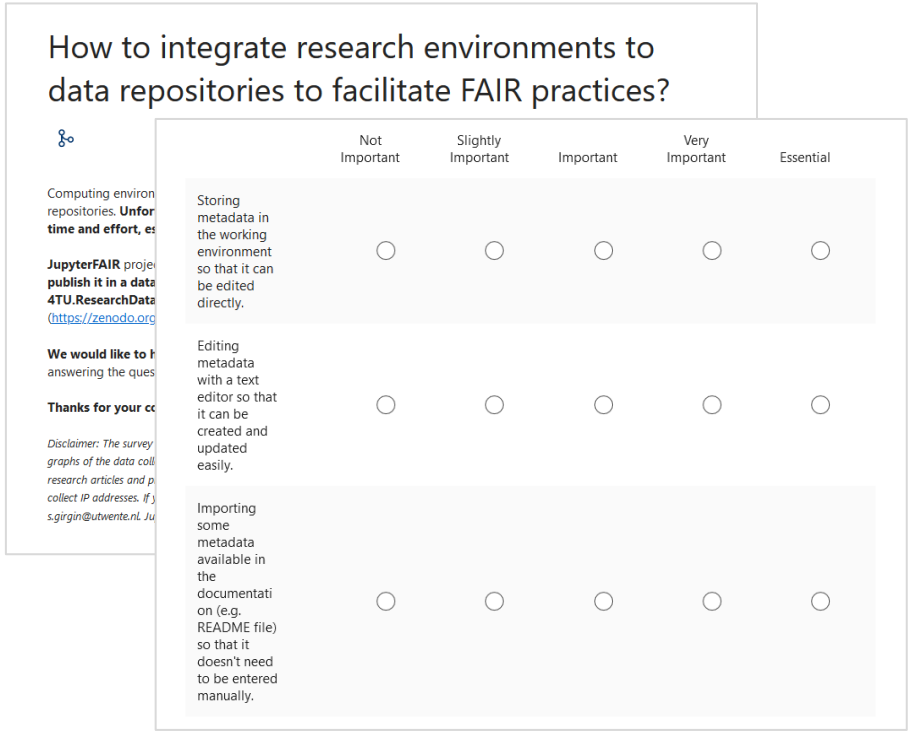
  Take part in the co-development effort with your programming and writing skills to improve code and documentation.

- **Visibility**

  Promote the tools if you find them useful.

https://storyset.com

You can start to **contribute now** by filling a short 5-min survey on research data publishing practices



How to integrate research environments to data repositories to facilitate FAIR practices?

Computing environ
repositories. **Unfor**
time and effort, es

JupyterFAIR proje
publish it in a data
4TU.ResearchData
(https://zenodo.org

We would like to h
answering the ques

Thanks for your co

Disclaimer: The survey
graphs of the data coll
research articles and p
collect IP addresses. If y
s.girgin@utwente.nL Ju

	Not Important	Slightly Important	Important	Very Important	Essential
Storing metadata in the working environment so that it can be edited directly.	○	○	○	○	○
Editing metadata with a text editor so that it can be created and updated easily.	○	○	○	○	○
Importing some metadata available in the documentation (e.g. README file) so that it doesn't need to be entered manually.	○	○	○	○	○

https://forms.office.com/r/Xg7RqwsTiS

Check our online resources to **learn more** about the tools

Code Repositories

https://github.com/ITC-CRIB/fairly

https://github.com/ITC-CRIB/jupyter-fairly

Watch the repositories for new features and fixes!

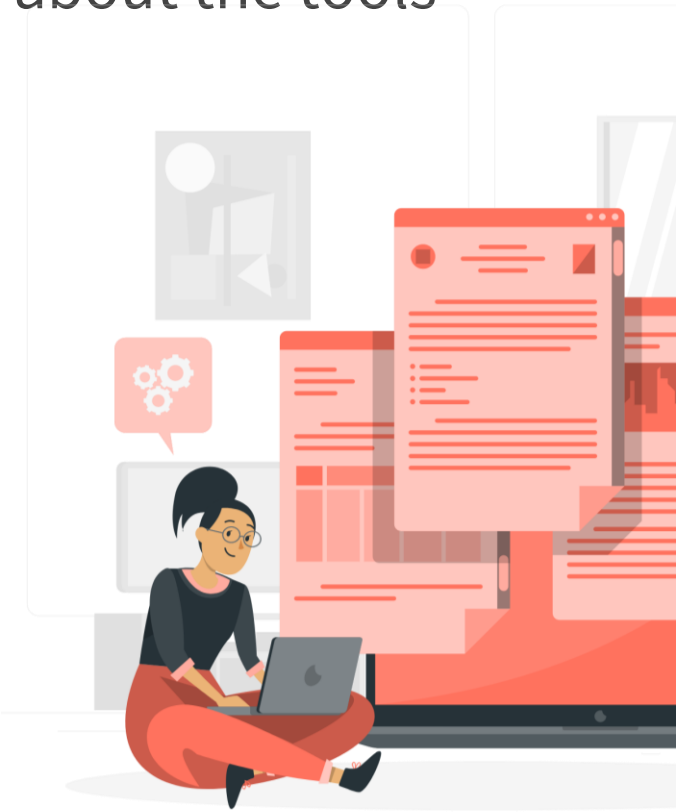
User Documentation

https://fairly.readthedocs.io/en/latest

Package Distributions

https://pypi.org/project/fairly

https://pypi.org/project/jupyter-fairly



https://storyset.com

Thanks for your time!

Please contact us for further **questions** or **training requests**:

Dr. Ing. Serkan Girgin MSc
s.girgin@utwente.nl

Manuel Garcia Alvarez
m.g.garciaalvarez@tudelft.nl

José Carlos Urra Llanusa
j.c.urrallanusa@tudelft.nl

https://twitter.com/JupyterFAIR*

Follow us to get informed on new features and events!