Git & Open Science

Matthieu Haefele

Pau, October 2023

Why bothering with open science ?

- Because data generated by any research project funded by the state by more than 50% MUST be openly accessible by law since 2016 !
- To access data published in papers and reuse them easily

FAIR principles

- **F**indable
- **A**ccessible
- **I**nteroperable
- **R**eusable

It applies on three objects:

- the data itself
- the metadata that describes the data
- the infrastructure that stores both the (meta)data

Findable

The first step in (re)using data is to find them. Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, so this is an essential component of the FAIRification process.

- F1. (Meta)data are assigned a globally unique and persistent identifier
- F2. Data are described with rich metadata (defined by R1 below)
- F3. Metadata clearly and explicitly include the identifier of the data they describe
- F4. (Meta)data are registered or indexed in a searchable resource

Accessible

Once the user finds the required data, she/he/they need to know how they can be accessed, possibly including authentication and authorisation.

- A1. (Meta)data are retrievable by their identifier using a standardised communications protocol
  - A1.1 The protocol is open, free, and universally implementable
  - A1.2 The protocol allows for an authentication and authorisation procedure, where necessary
- A2. Metadata are accessible, even when the data are no longer available

Interoperable

The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.

- I1. (Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.
- I2. (Meta)data use vocabularies that follow FAIR principles
- I3. (Meta)data include qualified references to other (meta)data

Reusable

The ultimate goal of FAIR is to optimise the reuse of data. To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.

- R1. (Meta)data are richly described with a plurality of accurate and relevant attributes
  - R1.1. (Meta)data are released with a clear and accessible data usage license
  - R1.2. (Meta)data are associated with detailed provenance
  - R1.3. (Meta)data meet domain-relevant community standards

FAIR applied to data

- **F**indable: Digital Object Identifier (DOI) + data warehouse (e.g. datagouv.fr)
- **A**ccessible: ???
- **I**nteroperable: Metadata with Dublin Core format
- **R**eusable: etalab licence

But a decent Data Management Plan is the first step !

FAIR applied to source code

- **F**indable
  - Software Heritage Identifier (SWHID)
  - HAL catalog
- **A**ccessible
  - Software Heritage platform
  - HTTPS protocol, either tarball or web API

FAIR applied to source code

- **I**nteroperable
  - Standard programming language
  - Standard API of libraries
  - git as a mean to exchange source code
- **R**eusable
  - Open source / free software licence
  - Reproducibility of the software environment required by your application with tools like guix or nix

Open source / free software

Free software embeds an ethic on the four freedoms for software

1. Freedom to run the software
2. Freedom to distribute the software
3. Freedom to study and change the software
4. Freedom to distribute modified versions of the software

Actually point 4 is an obligation.

- Free software used to build yours => your software is also free software
- Then MUST be distributed under the same terms.

Open source / free software philosophy

Free software

- Software freedom translates to social freedom
- To guarantee the freedom of the user in society, freedom has to be imposed on the software

Open source software

- no ethic beyond the software development process
- Freedom is not an absolute concept
- Freedom should be allowed, not imposed

Open source / free software licences

- Without license the copyright applies (droit d'auteur en France): even if publicly accessible, nobody can do anything with your code without your permission
- Two types of licences
  - BSD, MIT, Cecill-B: **without copyleft**, i.e. you can do whatever you want with code, sell it, close it... Some other licences can be very restrictive
  - GNU GPL, Cecill: **with copyleft**, i.e. you must put any modified version under the same licence

This site can help you choose a licence

Create the metadata file

- There are several metadata formats for software
- HAL decided to use codemeta
- This file generator can help at creating the file to add to your repository

Archive on Software Heritage

If your code is under git

- Install the web browser SWH plug-in
- Click the button !

Summary

In order to have a FAIR code, i.e. open science ready

1. Put your code under git
2. (Develop and execute it in a reproducible environment provided by nix or guix)
3. Add an open source / free software licence
4. Add metadata information
5. Archive it on Software Heritage
6. Create en entry on HAL with the SWHID

Last words on software quality

- **Unit tests**: single function
- **Integration tests**: combination of functions
- **Functional / non regression tests**: full application

Software quality is strongly improved when functionalities are tested and those tests are executed regularly, typically at each commit.

Continuous integration feature of gitlab helps at doing this