FAIR & Research Software

Dr. Anna-Lena Lamprecht (@al_lamprecht)
Utrecht University, Netherlands

Dr. Salvador Capella-Gutierrez (@sj_capella)
Barcelona Supercomputing Center, Spain

CSDMS Webinar, 6 May 2021

Outline

Recap: The FAIR data principles

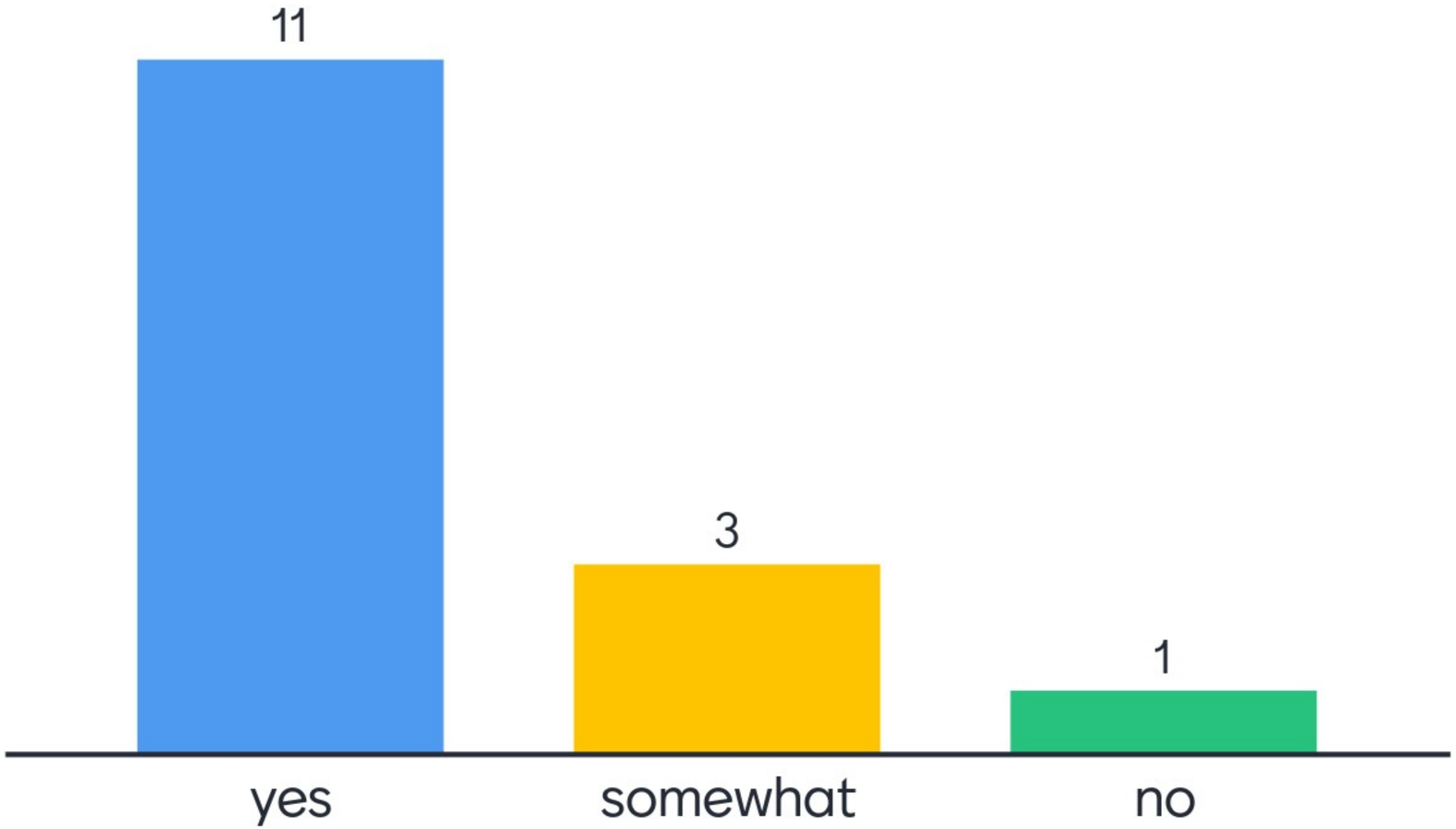
Data and software

Towards FAIR principles for research software

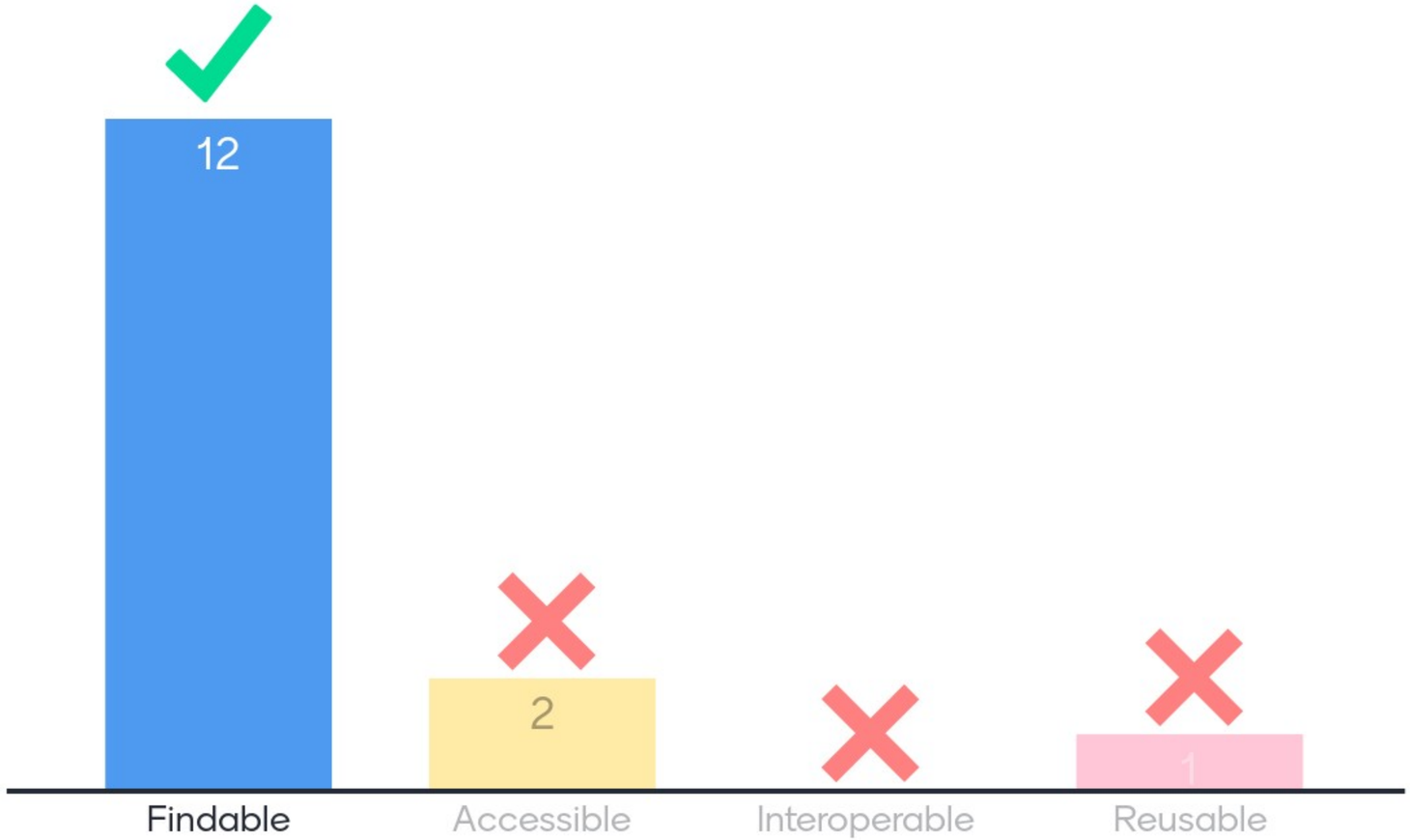
Outcomes so far

FAIR4RS community and ways to get involved

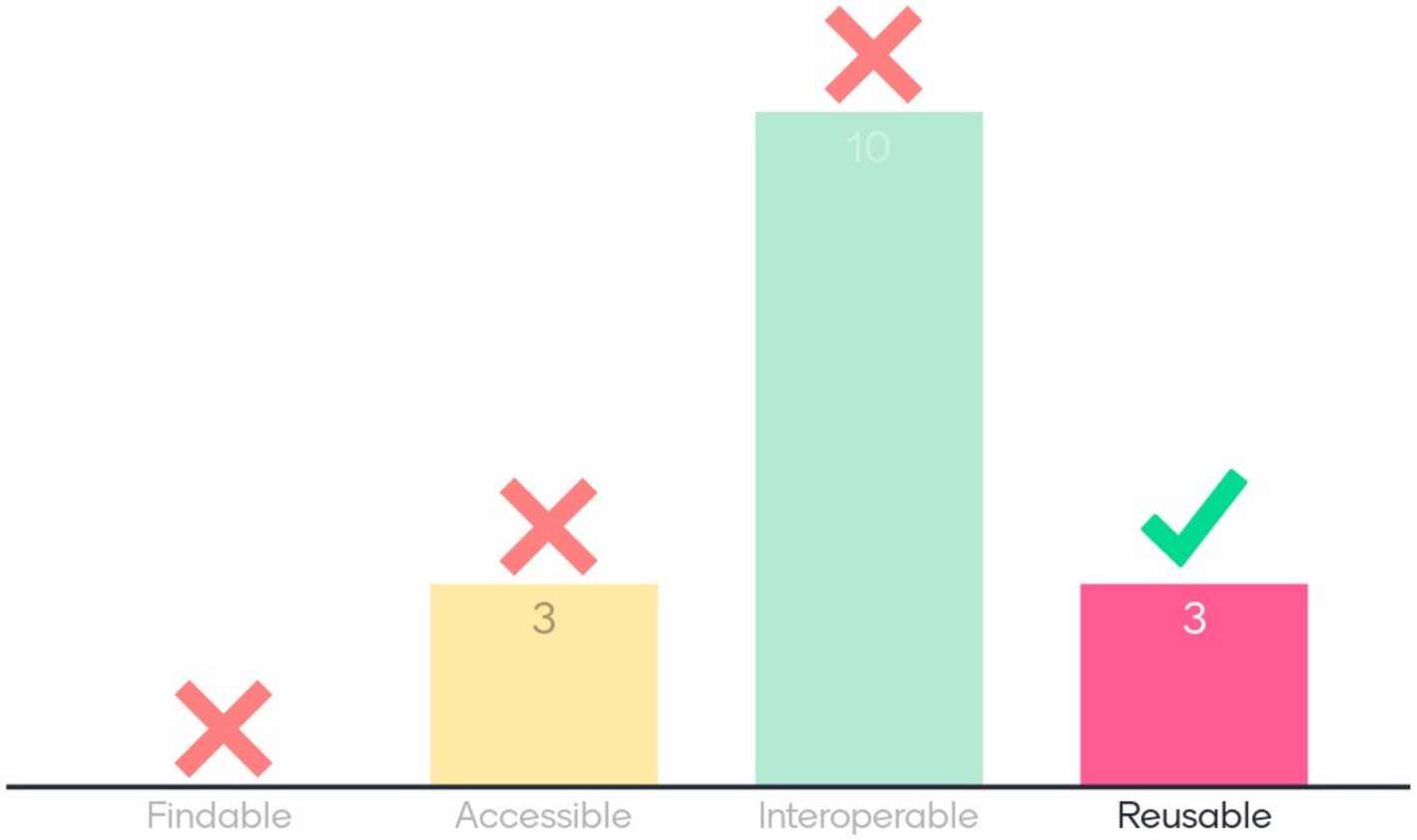
Are you familiar with the FAIR data principles?



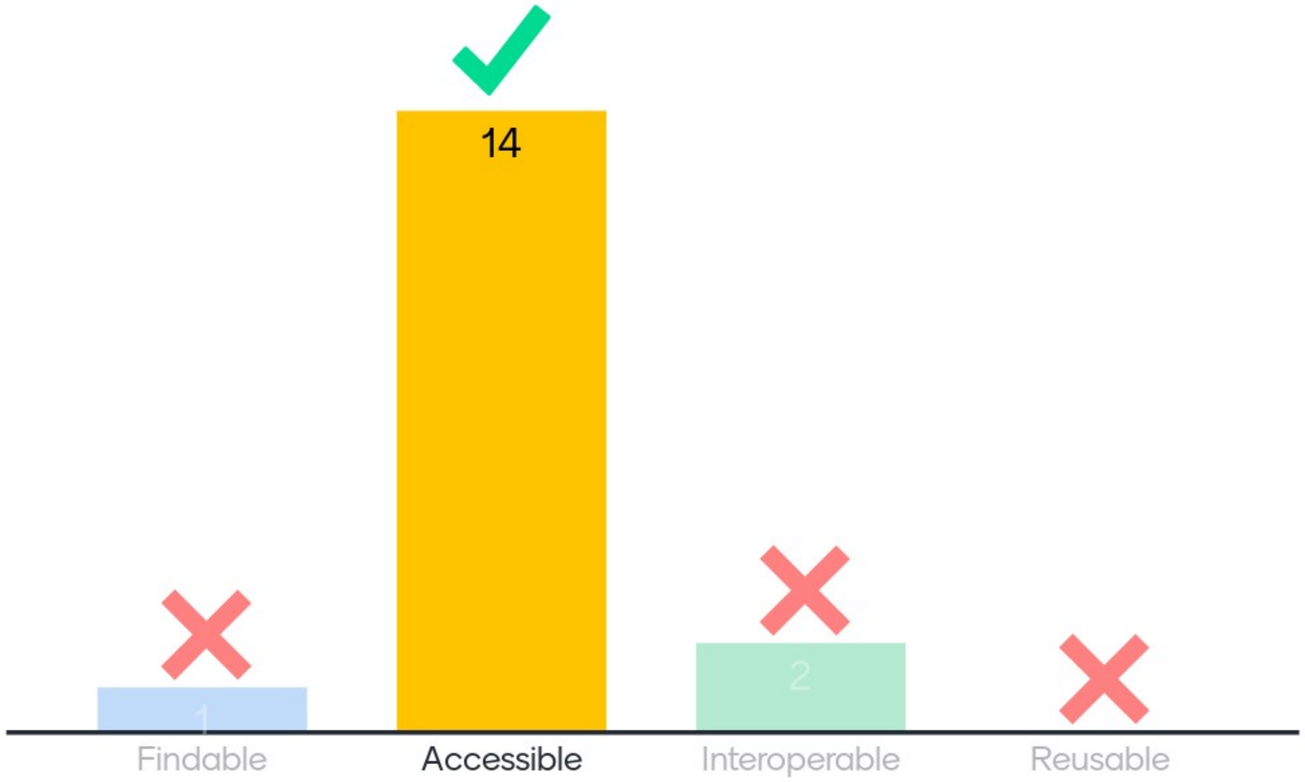
"(Meta)data are assigned a globally unique and persistent identifier." Which FAIR principle?



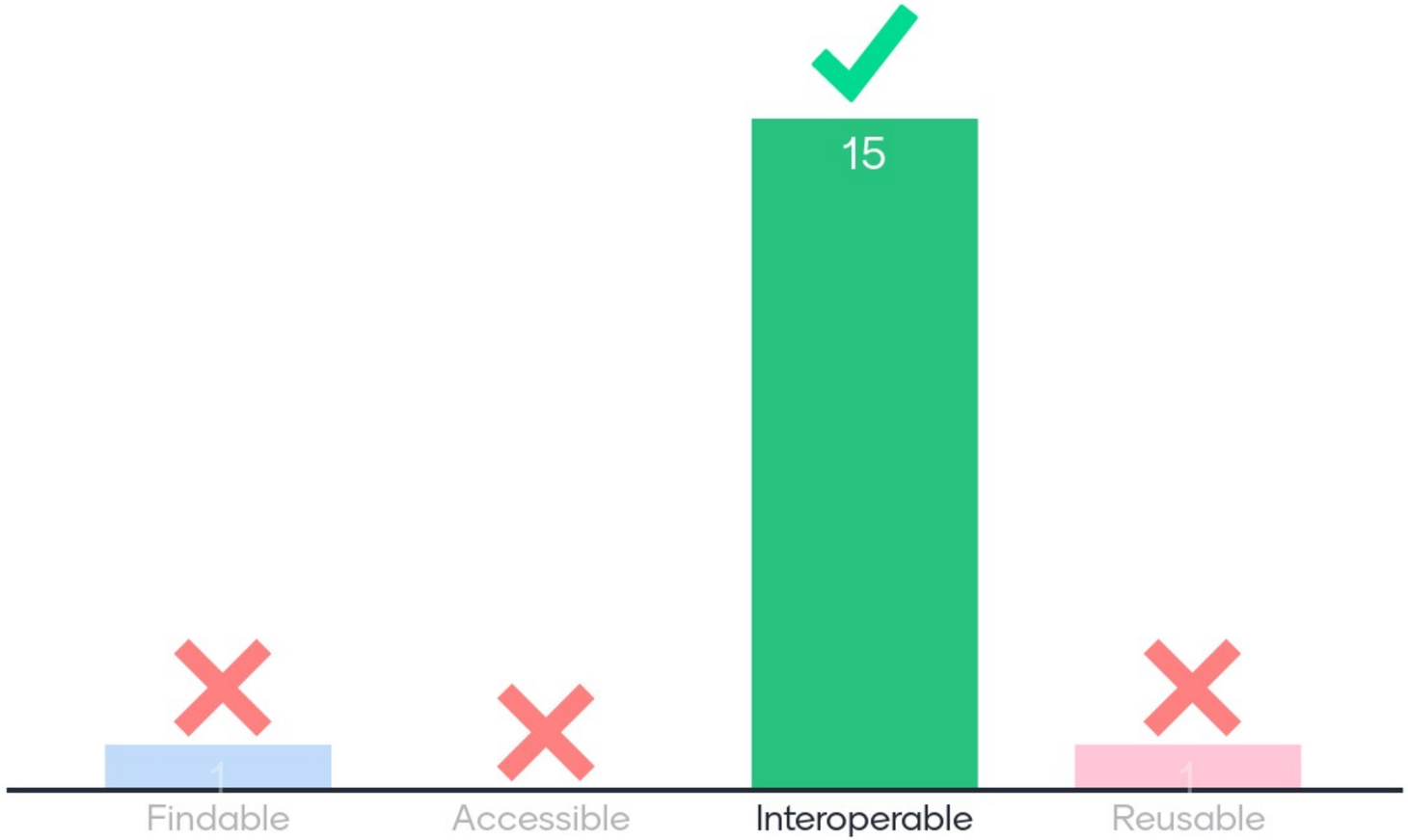
"(Meta)data meet domain-relevant community standards." Which FAIR principle?



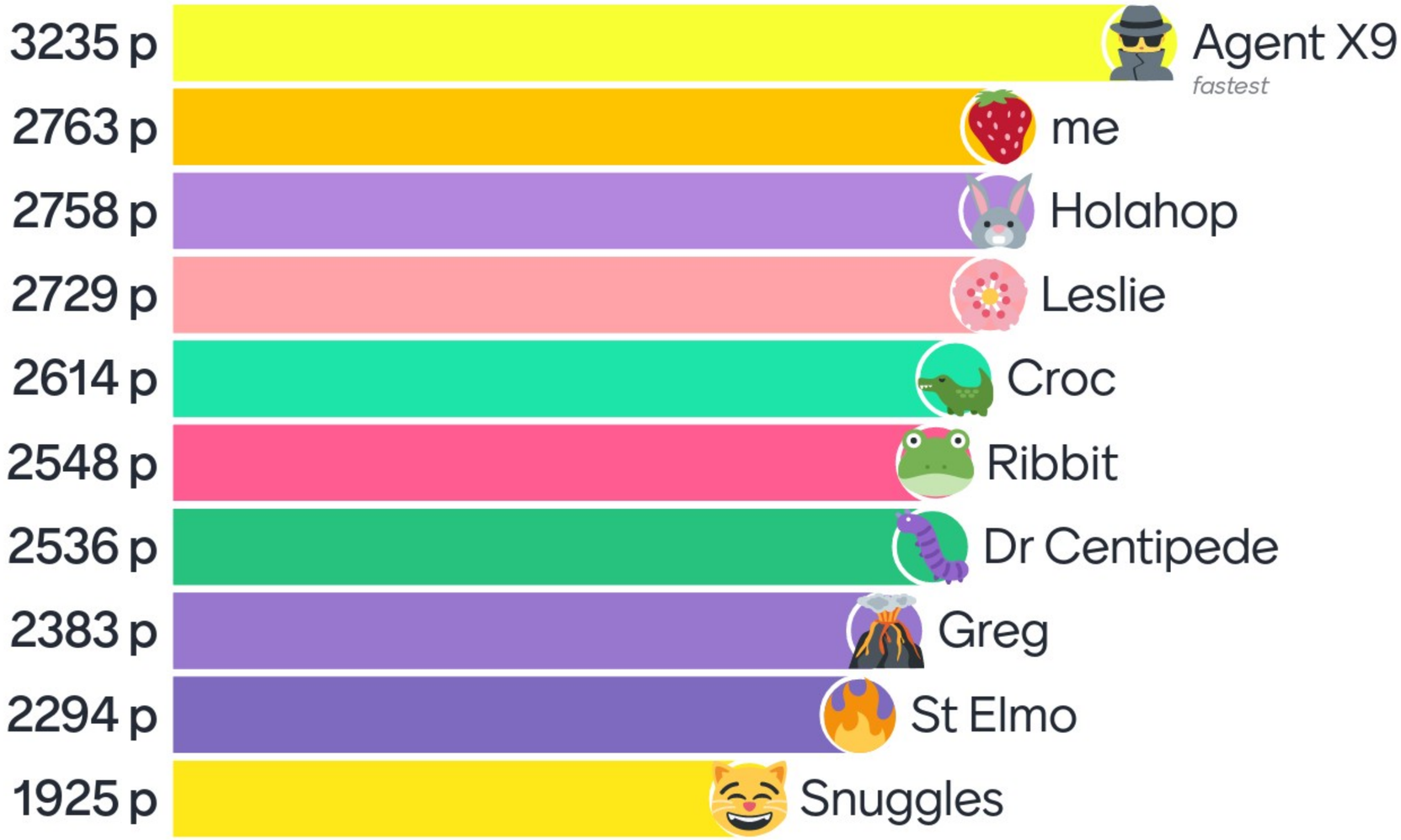
"(Meta)data are retrievable by their identifier using a standardised communications protocol." Which FAIR principle?



"(Meta)data include qualified references to other (meta)data." Which FAIR principle?



Leaderboard



FAIR

Findable
Accessible
Interoperable
Reusable

2016: "**The FAIR guiding principles for scientific data management and stewardship**" (Wilkinson et al., doi:10.1038/sdata.2016.18)

2018: "Central to the realization of FAIR are **FAIR Digital Objects, which may represent data, software or other research resources**." (European Commission)

But: **How do the FAIR Principles relate to software?**

Mismatch between the broad intentions of the **4 foundational FAIR principles** and how the **15 FAIR Guiding Principles** are communicated and perceived.

FAIR and software

An ongoing discussion...

Milestone:
"Towards FAIR Principles for Research Software", Lamprecht et al., Data Science, Vol.3, Iss.1, pp. 37–59, 2020, https://doi.org/10.3233/DS-190026

Most viewed article in Data Science in the first half of 2020!

Let's get into some of the thoughts and ideas in there.

What is data?

Recorded information

observationsmeasurements

Observations collected by humans or sensors

Information in it's raw-est form

Measured values of phenomena

information in a communicable form

Information in numerical or other forms

Recorded information

What is software?

Computer code written to accomplish a purpose

Code applied to data

Computer code composed in a human-readable programming language

Code packages to perform some operation

tool built to support a process, sometimes data collection

methodology

Tools based on a computer used to perform specific tasks, perhaps involving the use/output of data, but not necessarily

Scientific software: a set of equations, captured in a language to mimic a process

computer program structured to receive digital information and perform tasks

What is software?

source code and its resultant executable application

source code that required computational dependecies

What do data and software have in common?

software is data

Part of scientific process

Used together to solve a problem

Both are stored on a computer

Both are forms of information; like computer code, most data today exist in digital form.

must have standard to work together

they support research outputs and are very valuable

data have input and output that related to software.software is computational engine to use data

Both can have multiple versions

What do data and software have in common?

use interpreted language to communicate

The need to be FAIR

What makes data and software different?

Software without data aren't too useful (for science)

how they accumulate

data is understandable using metadata.Software can't understand using metadata itself

Data are not executable

Software can be re-run many times, data observations aren't repeated exactly

they require different knowledge to create, implement, document

Software needs data to work

data doesn't change and software can be versioned

data is independent of a function, execution, an operable structure

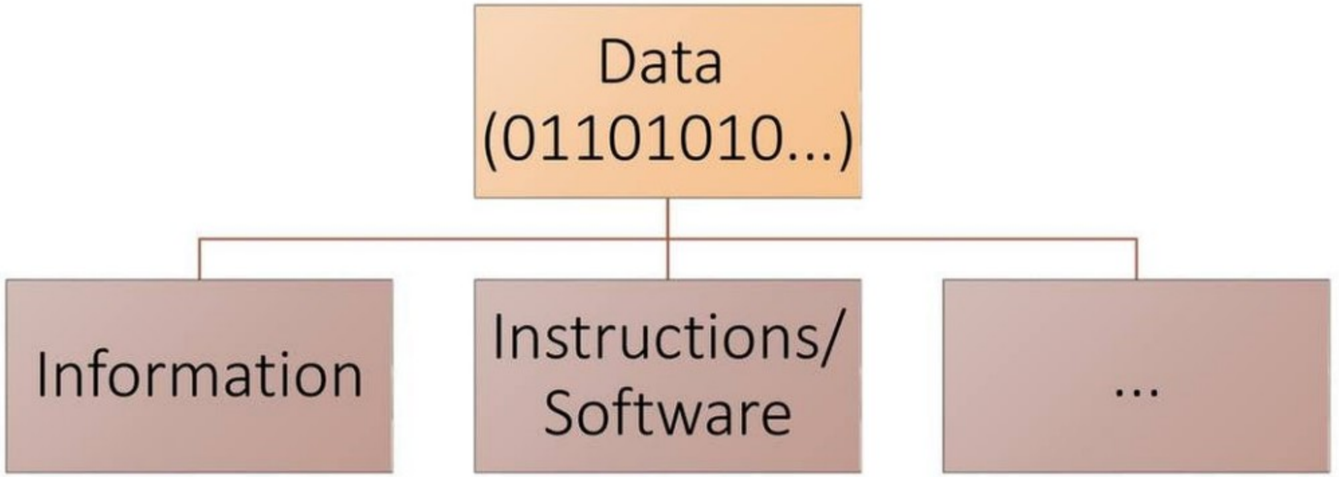
What makes data and software different?

different expertise is needed to create each

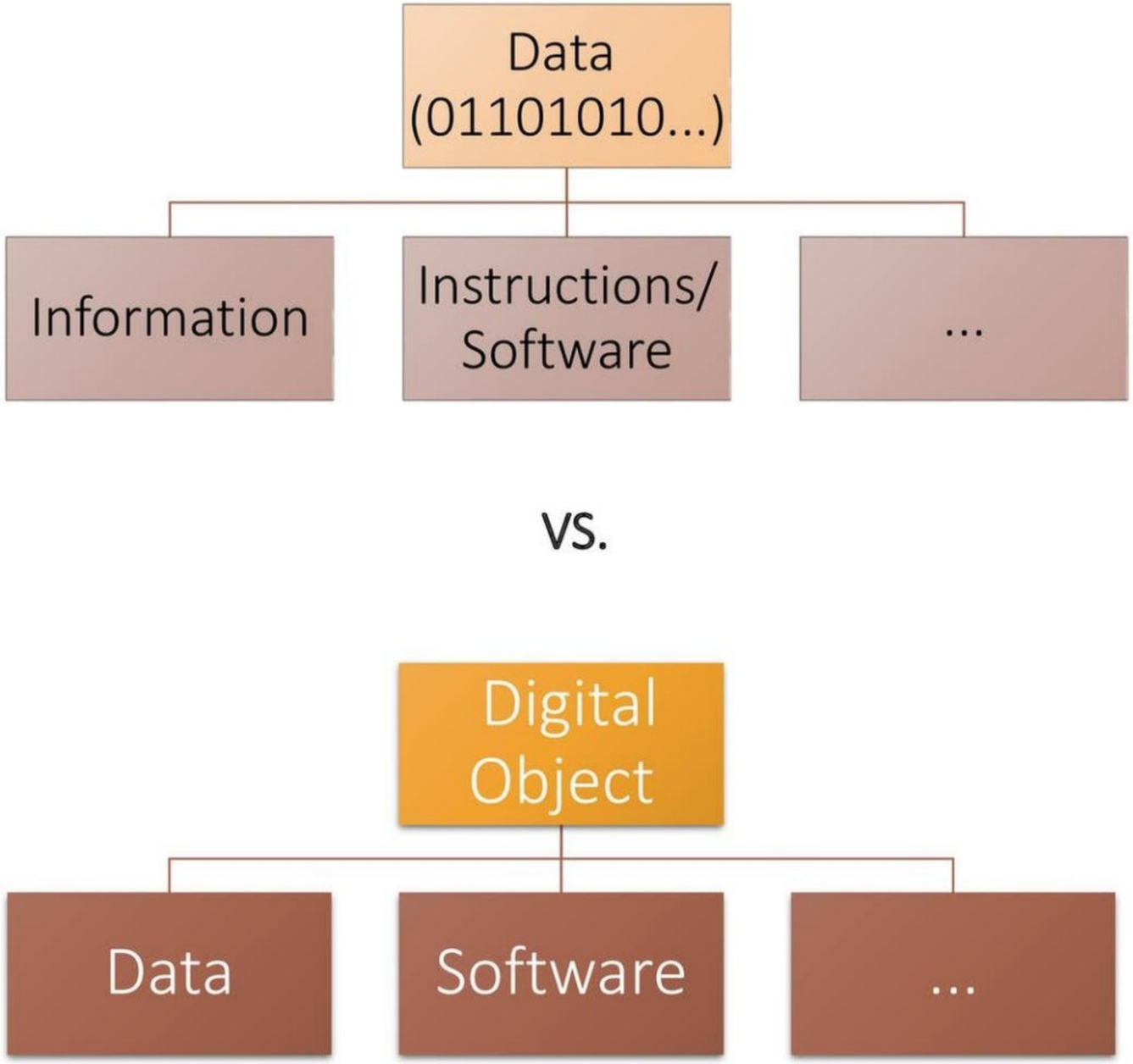
Data represent, whereas software instructs

Ideally - provide use cases for when not to make it open?

Software is (not) data



Software is (not) data



Research software

Research software is "**software that is used to generate, process or analyze results that you intend to appear in a publication**" (Hettrick et al., 2014).

Many forms.

Many purposes.

Many distribution channels.

Traditionally, often created as Free and/or Open Source Software (FOSS).

FAIR and FOSS

Clear overlap of objectives, but not the same.

FOSS: Open source code, open licenses.

FAIR: Open data not a requirement.

- Due to, e.g., privacy and sensitivity concerns with patients' health records.
- Not in the same way valid for research software.
- There is even a demand to make methods available!

Should FAIR software require FOSS?

Ongoing discussion ...

What do you think? Should FAIR require software to have an open license? Why, or why not?

Yes, on the principle of Accessibility

In some cases would be a benefit.

both are integral parts of the Open Science, Open Data, Open Source

No, because sometimes you are constrained to use some specific software that is not open

Yes, it is the default.

Should contain a strong recommendation towards an open license. At least have an executable open available (but code only recommended

Open license by default, which exceptions allowable with good justification

Some commercial software is very useful for science - so perhaps providing instructions to repeat what you did that someone with a license should be able to execute

no, there are multiple ways to execute a program, requirement would lead to community adoption of a program as the only method, and limit progress

What do you think? Should FAIR require software to have an open license? Why, or why not?

No, quality assessment is a subjective issue

FAIR and software quality

Software quality is a major concern in RSE.

Can FAIR meet the expectations?

Distinguish between **form** and **function** of software:

- Quality of the **form** of software can be covered by FAIR (code quality, maintainability).
- Quality of the **functionality** of software goes beyond FAIR (functional correctness, software security, computational efficiency).

Should FAIR also take content quality into account? Why (not)?

There is no "Q" in FAIR...

No, because different scientific fields have different standards for accuracy /quality so blanket statements are difficult

Quality is included implicitly, e.g. in R - reusable software should have undergone code review to ensure "correctness"

No - I'm not a computer scientist, but the code I write works for my analyses even if it isn't pretty

.....Ahh, I don't know..... No?

Tangential, but knowing your code is used by others may improve its quality

No, that's a different issue.

Quality may be described as part of a metadata

No, perhaps just structural conformance that helps domain specific reviewers assess quality more easily

FAIR4RS WG

RDA FAIR for Research Software (FAIR4RS) WG
https://www.rd-alliance.org/groups/fair-research-software-fair4rs-wg

Jointly convened as an RDA Working Group, FORCE11 Working Group, and Research Software Alliance (ReSA) Taskforce.

First draft of FAIR Principles for Research Software presented on 21 April 2021,
see https://www.researchsoft.org/news/2021-04/

Draft FAIR Principles for Research Software

Findable: The software, and its associated metadata, should be easy to find for both humans and computers.

F1. Software is assigned a globally unique and persistent identifier that supports assigning of versions

F2. Software is described with rich metadata to support search and discoverability

F3. Metadata clearly and explicitly include the identifier of the software they describe

F4. Software is registered or indexed in a searchable resource

Accessible: The software, and its metadata, must be retrievable via standardized protocols.

A1. Software is retrievable by its identifier using a standardized communications protocol

- A1.1. The protocol is open, free, and universally implementable
- A1.2. The protocol allows for an authentication and authorization procedure, where necessary

A2. Metadata are accessible, even when the software is no longer available

Interoperable: The software interoperates with other software through exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs).

I1. Software reads, writes and exchanges data in a way that meets domain-relevant community standards

I2. Software includes qualified references to other objects

Reusable: The software is both usable (it can be executed) and reusable (it can be understood, modified, built upon, or incorporated into other software).

R1. Software is richly described with a plurality of accurate and relevant attributes

- R1.1. Software is made available with a clear and accessible license
- R1.2. Software is associated with detailed provenance

R2. Software includes qualified references to other software

R3. Software meets domain-relevant community standards

FAIR Metrics

Another ongoing discussion.

How to measure software FAIRness?

Introduction: OpenEBench Technical Monitoring

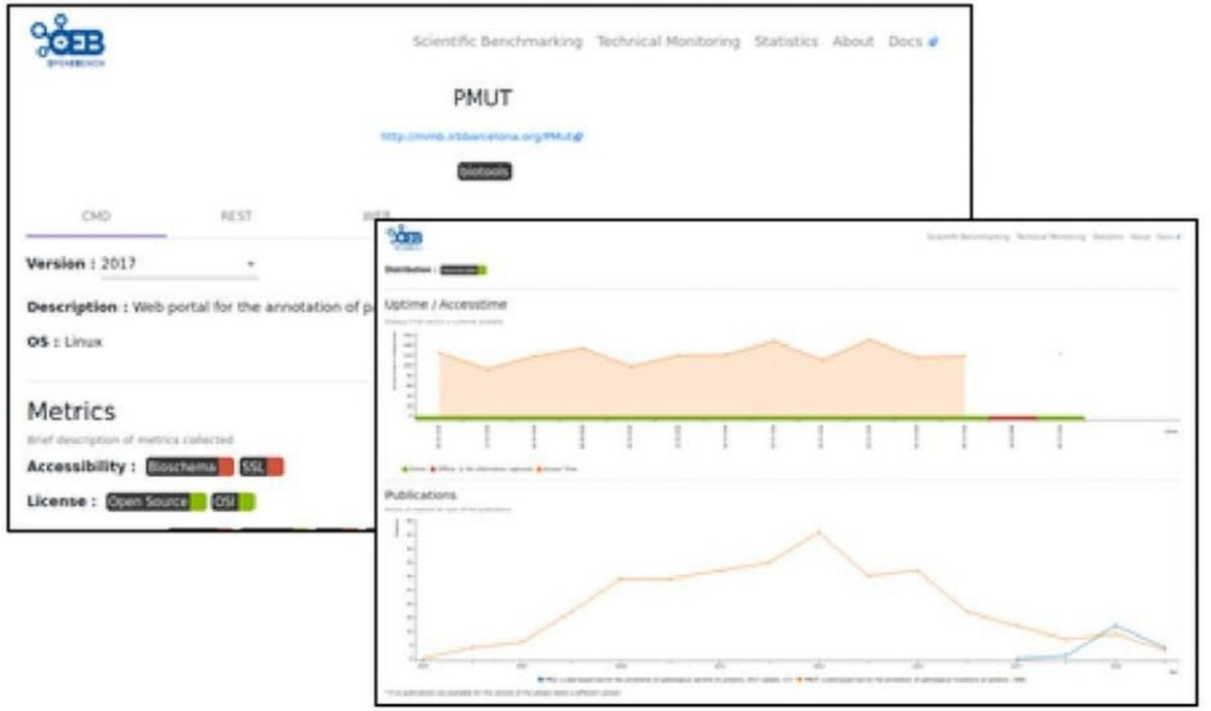
At OpenEBench, we aim to provide and monitor **technical details** and **quality indicators** of bioinformatic tools.

- For **individual tools** to aid selection by scientists along with benchmarking results.

  OpenEBench technical monitoring

- For the **whole population** of tools.

  OpenEBench Tools Observatory



Introduction: Software Quality Monitoring

Quality assessment framework

Theoretical collective effort

- *Towards FAIR principles for research software*
- Software Management Plan.
- Workshops at ELIXIR All Hands.
- Community-led discussions by RDA, FORCE11 and ELIXIR on how to effectively apply FAIR principles to research software.

Infrastructure

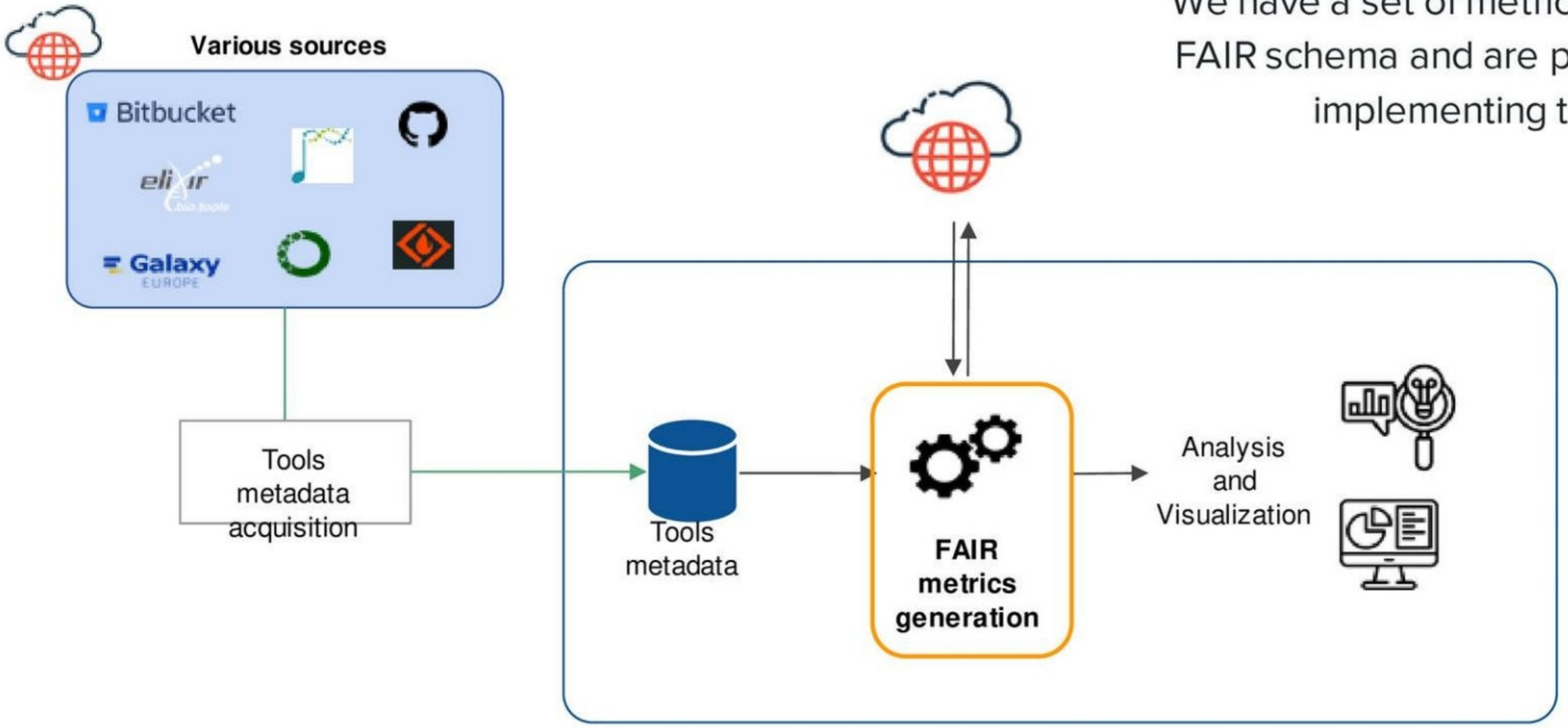
Technical

- ETL strategies
- Metrics generation tools
- Platform to release results

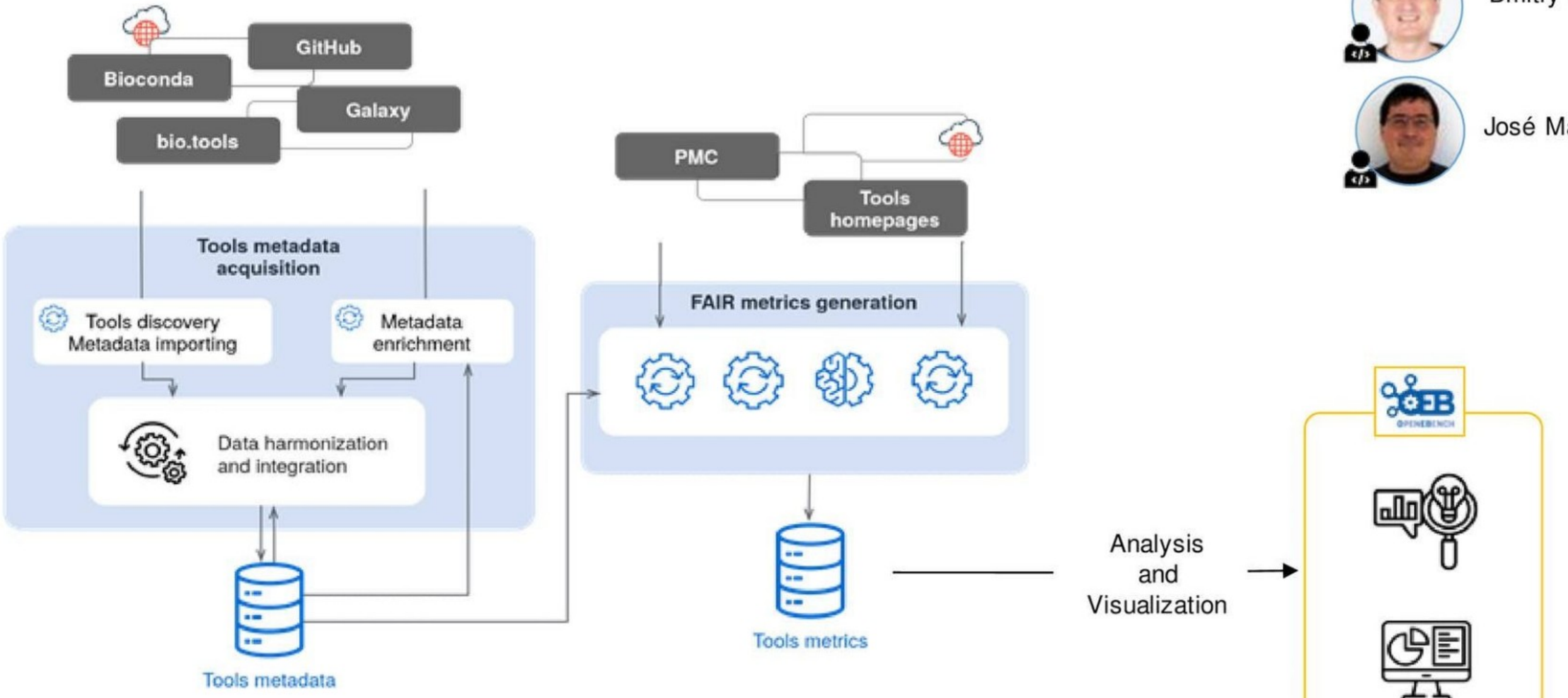
OPENEBENCH

Data Extraction and Transformation: Overview

We have a set of metrics following a FAIR schema and are progressively implementing them



Data Extraction and Transformation: Pipeline

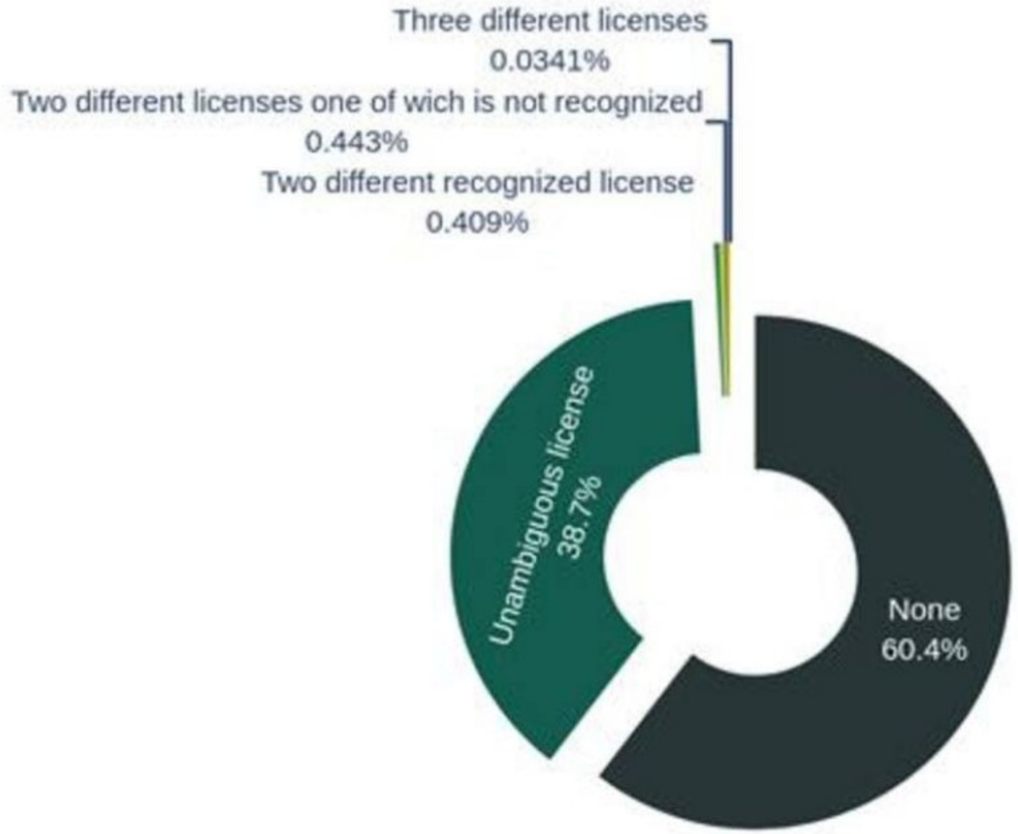


Dmitry Repchevski

José María Fernández

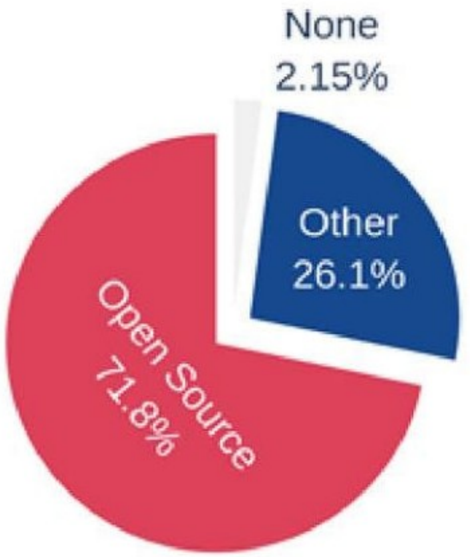
Results: Licensing

In some cases the licenses obtained from different sources are not coherent:



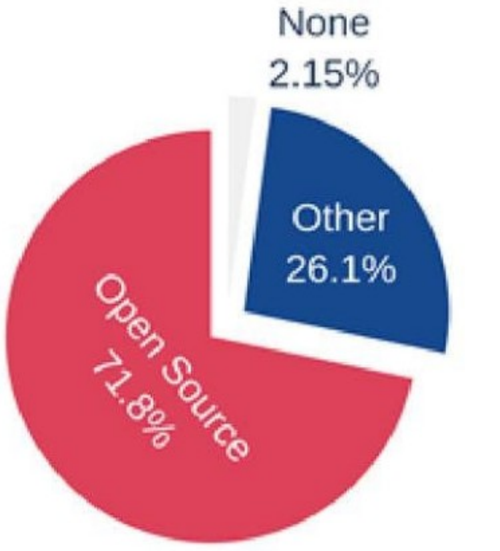
Licensing is one of the most crucial features of a piece of software, determining both its **Accessibility** and **Reusability**

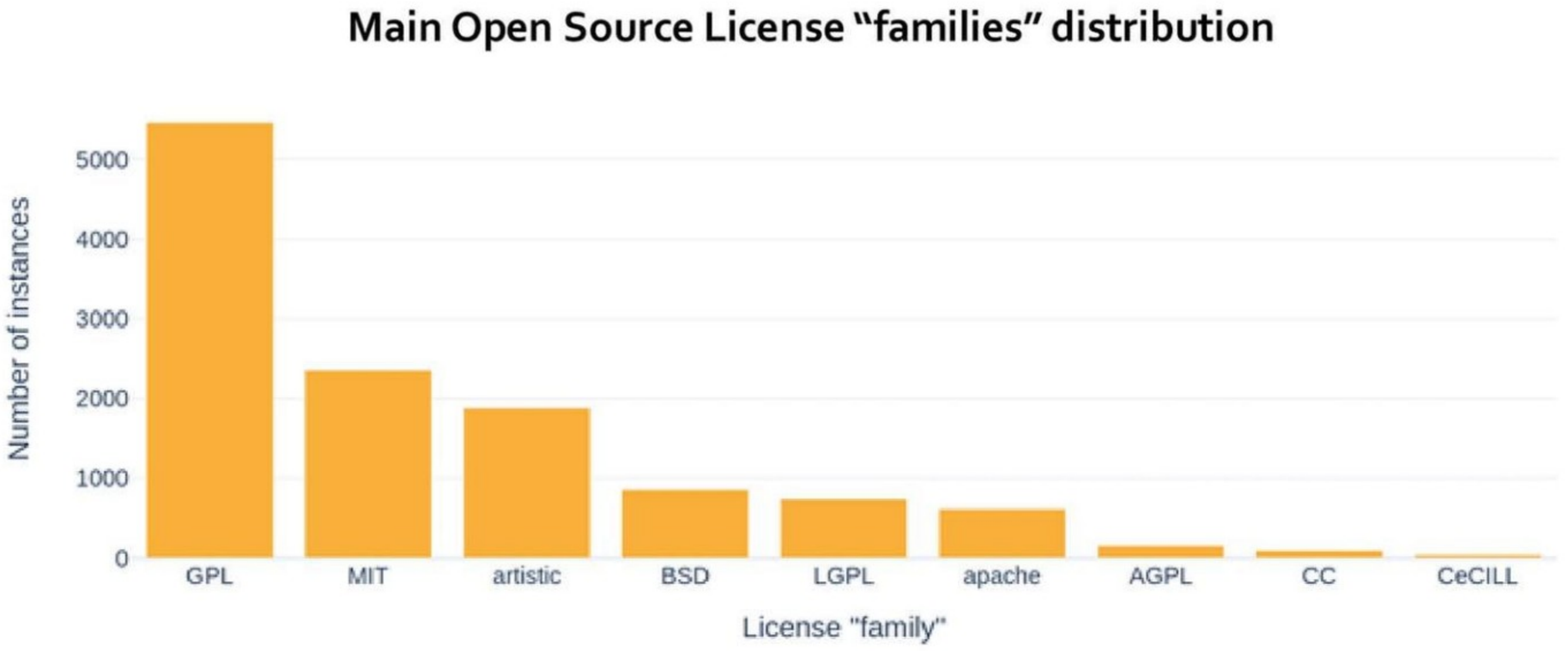
Among the **unambiguous** licenses we find



If a license is NOT stated, the software CANNOT be used!!

Results: Licensing

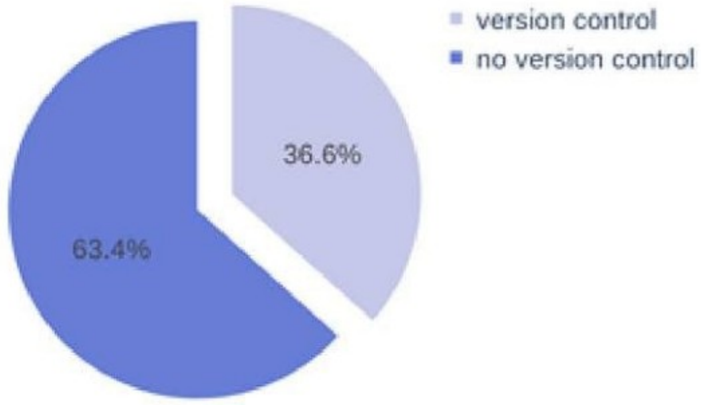


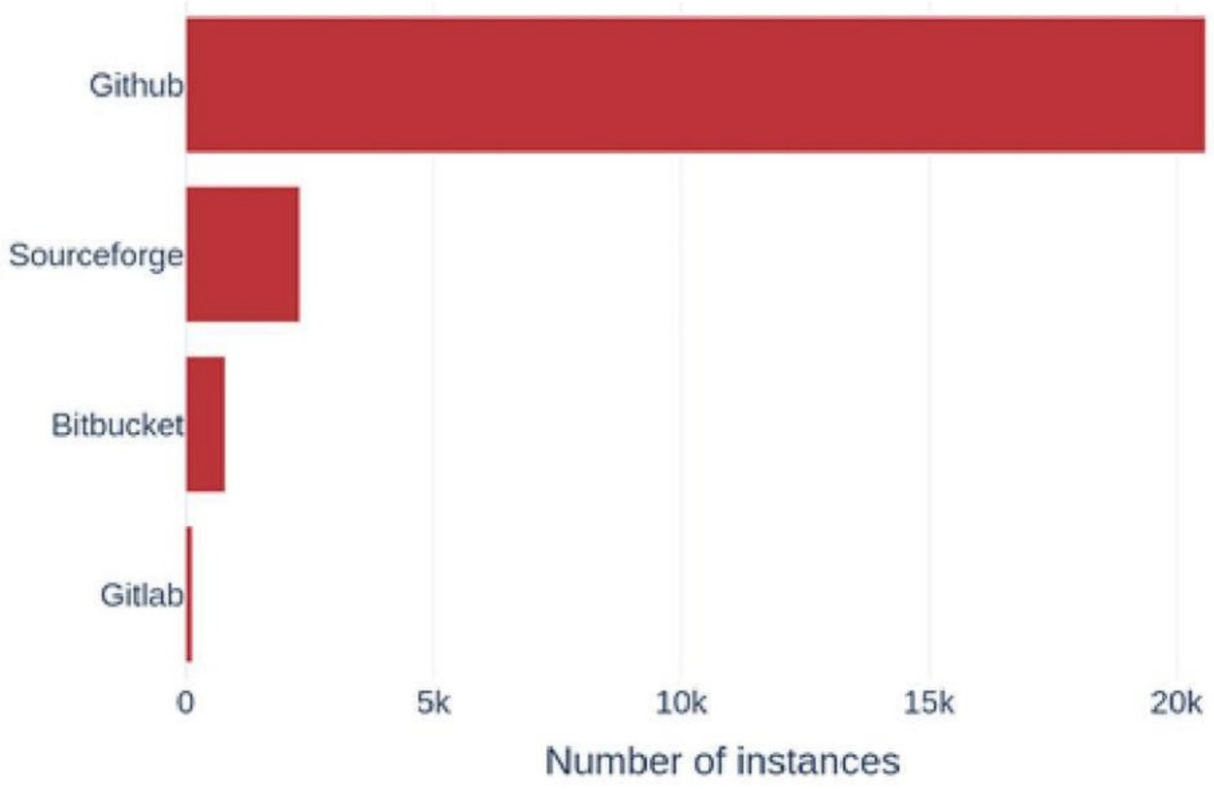


Results: Version Control

Version control offers a standardized record of source code changes, making it easier to be **Reused**.

The main version control systems also allow straightforward **Accessibility**





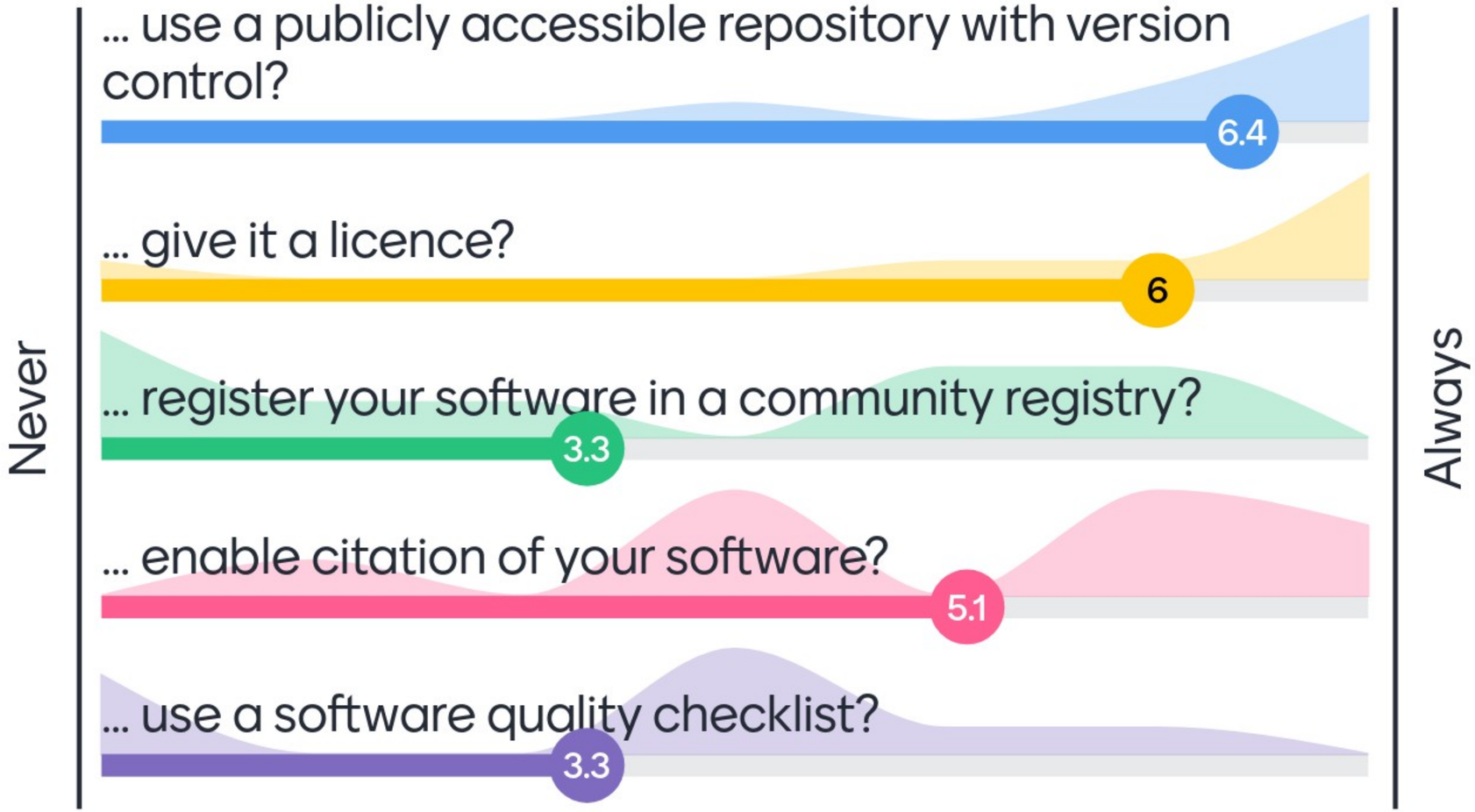
Practical FAIR

"Five Recommendations for FAIR Software",
Netherlands eScience Center, https://fair-software.eu/

1. Use a publicly accessible repository with version control.
2. Add a license.
3. Register your code in a community registry.
4. Enable citation of the software.
5. Use a software quality checklist.

(Your organization can endorse this!)

How about your software? Do you...



Incomplete FAIR software reading list

"Top 10 FAIR Data & Software Things", Martinez et al., Zenodo, 2019, https://doi.org/10.5281/zenodo.3409968

"Towards FAIR Principles for Research Software", Lamprecht et al., Data Science, 2020, https://doi.org/10.3233/DS-190026

"FAIR Computational Workflows", Goble et al., Data Intelligence, 2020, https://doi.org/10.1162/dint_a_00033

"From FAIR research data toward FAIR and open research software", Hasselbring et al., Information Technology, 2020, https://doi.org/10.1515/itit-2019-0040

FAIRsFAIR "Assessment report on 'FAIRness of software'", Gruenpeter et al., Zenodo, 2020, https://doi.org/10.5281/zenodo.4095092

Acknowledgments

To the numerous people who contributed to the discussions around FAIR research software at different occasions and keep the work going!

Thank you!