

Software Pillar of Open Science

challenges and opportunities

Roberto Di Cosmo

Director, Software Heritage
Inria and Université de Paris Cité

April 2025



Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

- 1 Introduction
- 2 Software and Source Code
- 3 France is leading the way
- 4 Addressing the ARCD needs
- 5 Adoption indicators and call to action

Short Bio: Roberto Di Cosmo

Computer Science professor in Paris, now working at INRIA

- 35+ years of research (Theor. CS, Programming, Software Engineering, Erdos #: 3)
- 25+ years of Free and Open Source Software
- 15+ years building and directing structures for the common good



1999 *DemoLinux* – first live GNU/Linux distro

2007 *Free Software Thematic Group*

150 members 40 projects 200Me

2008 *Mancoosi project* www.mancoosi.org

2010 *IRILL* www.irill.org

2015 *Software Heritage* at INRIA

2018 *National Committee for Open Science*, France

2021 *EOSC Task Force on Infrastructures for Software*,
European Union

- 1 Introduction
- 2 Software and Source Code
- 3 France is leading the way
- 4 Addressing the ARCD needs
- 5 Adoption indicators and call to action

Why Open Science?

Open Science (Second National Plan for Open Science, France, 2021)

Unhindered dissemination of results, methods and products from scientific research. It draws on *the opportunity provided by recent digital progress* to develop *open access to publications* and – as much as possible – *data, source code and research methods*.

Jean-Eric Paquet (EU DGRI, on the objective of Open Science)

“Increase *scientific quality*, the *pace of discovery and technological development*, as well as *societal trust in science*.”

Mariya Gabriel (EU Commissioner for Research)

The COVID-19 crisis has also shown that cooperation at international level in research and innovation is more important than ever, including through *open access to data and results*. *No nation, no country can tackle any of these global challenges alone*.

Yuval Noah Harari (on COVID 19)

“*The real antidote [to epidemic] is scientific knowledge and global cooperation.*”

Software Source Code is Precious Knowledge

Harold Abelson, Structure and Interpretation of Computer Programs (1st ed.) 1985

“Programs must be written for people to read, and only incidentally for machines to execute.”

Apollo 11 source code (excerpt)

```
P63SP0T3      CA      BIT6      # IS THE LR ANTENNA IN POSITION 1 YET
              EXTEND
              RAND      CHAN33
              EXTEND
              BZF      P63SP0T4      # BRANCH IF ANTENNA ALREADY IN POSITION 1

              CAF      CODE500      # ASTRONAUT: PLEASE CRANK THE
              TC      BANKCALL      # SILLY THING AROUND
              CADR      GOPERF1
              TCF      GOTOP00H      # TERMINATE
              TCF      P63SP0T3      # PROCEED SEE IF HE'S LYING

P63SP0T4      TC      BANKCALL      # ENTER INITIALIZE LANDING RADAR
              CADR      SETPOS1

              TC      POSTJUMP      # OFF TO SEE THE WIZARD ...
              CADR      BURNBABY
```

Quake III source code (excerpt)

```
float Q_rsqrt( float number )
{
    long i;
    float x2, y;
    const float threehalfs = 1.5F;

    x2 = number * 0.5F;
    y = number;
    i = * ( long * ) &y; // evil floating point bit level hacking
    i = 0x5f3759df - ( i >> 1 ); // what the fuck?
    y = * ( float * ) &i;
    y = y * ( threehalfs - ( x2 * y * y ) ); // 1st iteration
    // y = y * ( threehalfs - ( x2 * y * y ) ); // 2nd iteration, this
    // can be removed

    return y;
}
```

Len Shustek, Computer History Museum

2006

“Source code provides a view into the mind of the designer.”

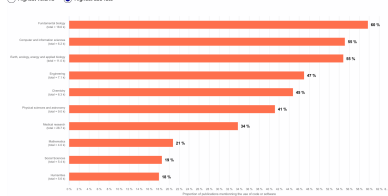
Software is a pillar of Open Science

Software powers modern research

Proportion of publications in France that mention the use of code or software by discipline

Sort by:

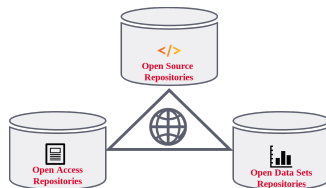
☐ Highest volume ☒ Highest use rate



Over 20% of articles using software across all disciplines share it

2024 French Open Science Monitor

Key pillar: software



Links are **important**

Nota Bene

software may be a *tool*, a *research outcome* and a *research object*

access to the *source code* is essential!

Preserving (the history of) source code is necessary for *reproducibility*

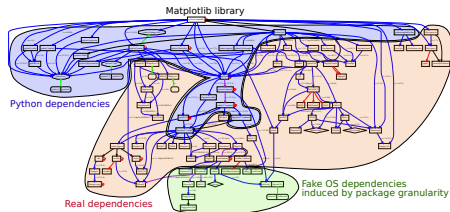
Source code is *special* (software is *not* data)

Software *evolves* over time

- projects may last decades
- the *development history* is key to its *understanding*

Complexity

- *millions* of lines of code
- large *web of dependencies*
 - easy to break, difficult to maintain
 - *research software* a thin top layer
- sophisticated *developer communities*

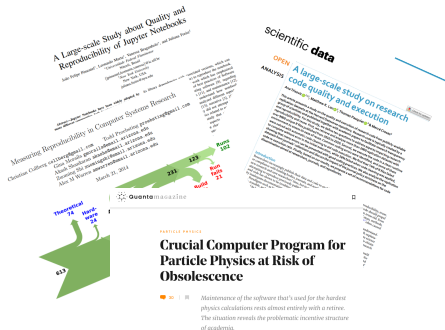


The human side

design, algorithm, code, test, documentation, community, funding
and so many more facets ...

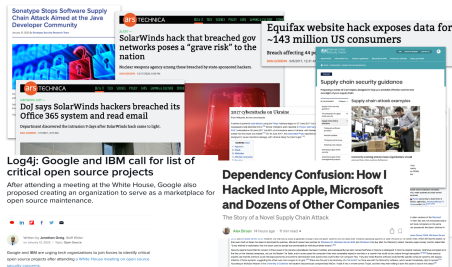
How are we managing our software ?

Reproducibility, maintenance in Academia



(articles: [here](#), [here](#), [here](#) and [here](#))

Security, integrity, traceability in Industry



Can they track the software that they

- ship, use, acquire
- has that bug or vulnerability

awareness is raising at the level of public policy

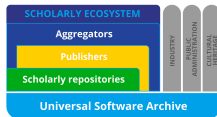
Paris Call on Software Source code (2019, UNESCO)

40 international experts call to “promote software development as a valuable research activity, and research software as a key enabler for Open Science/Open Research, [...] recognising in the careers of academics their contributions to high quality software development, in all their forms”



Open Source in UNESCO [recommendations](#) for Open Science, 2021

Software in the EOSC



2020 [EOSC SIRS](#) connect scholarly ecosystem via Software Heritage

2021 [EOSC Task Force](#) on Infrastructures for Research Software

2022 [FAIRCORE4EOSC project](#) WP6 implements SIRS report

And much more

Software track in [OSEC 2022](#), Software working group launched in Science Europe, DFG adds software [to model CV \(9/22\)](#), NASA unveils [Open Science policy \(12/22\)](#), ...

ARDC

- **Archive** for retrieval (*reproducibility*)
- **Reference** for identification (*reproducibility*)
- **Describe** for discovery and reuse
- **Cite/Credit** for credit and evaluation

Before ARDC

- **Development** practices and tools (VCS, build system, test suites, CI, code quality, ...)
- **Opening up** towards a community (documentation, organization, communication)

Need training, tooling, infrastructures, best practices

Beyond ARDC

- **Policies** (dissemination, reuse, careers, ...)
- **Sustainability** (legal, financial, etc.)
- Technology transfer
- Advanced technologies and tools (quality, traceability, etc.)

a humbling challenge, and a complex one (we are not in a vacuum)

- 1 Introduction
- 2 Software and Source Code
- 3 France is leading the way**
- 4 Addressing the ARCD needs
- 5 Adoption indicators and call to action

French National plan for Open Science, 2021-2024

SECOND FRENCH PLAN FOR OPEN SCIENCE

Generalising open science in France 2021-2024



Second French Plan for Open Science

Launch on 6 July 2021 by Frédérique Vidal, Minister for Higher Education, Research and Innovation

- Multiplying the **levers for change** in order to **generalise open science practices**
- Structuring the **policy for opening up or sharing research data**
- New commitments to the **opening of source code** produced by research
- **European and international inclusion** in the context of the French Presidency of the European Union
- **Disciplinary and thematic variations**: open science policies must be adapted to disciplinary specificities

GENERALISING
OPEN SCIENCE
IN FRANCE 2021-2024

Path Three : Opening up and promoting source code produced by research

7

Recognize and support the dissemination under an open source license of software produced by publicly funded research programmes

« The opening of software source code is a major challenge for the **reproducibility** of scientific results. »

8

Highlight the production of **source code** from higher education, research and innovation

« Distribution of software products under **open source licence** will be preferred. »

9

Define and promote an **open source software policy**

Define and promote an open source software policy

- Produce a **National Charter for Open Source Software** coming from higher education, research and innovation
- Develop the **link between data and software** through a network of **Chief Data Officers** in the various universities and research performing organisations.
- Develop the **economic models of open source software** and make them known within commercialization services
- **Support Software Heritage** and recommend it for the archiving and referencing of source code

Recognise source code as a contribution to research

- Create an **open source research software prize**
- **Provide greater recognition** for software production in the career of researchers, research support staff

Build an ecosystem that connects code, data and publications

- Develop **proper coordination** between software forges, open publication archives, data repositories and the scientific publishing sector.

Five action lines (see [details online](#))

- Identifying and highlighting research software production
- Technical and social tools and best practices
- Valorization and sustainability
- Liaison and animation at national, European, and international levels
- Recognition and careers

Source Code primer key concepts



- for students
- for teachers
- for researchers



Report on software forges in academia (FR):



- needs
- options
- limitations



Annual award *Establishing a national research software award.* Open Research Europe 2023



National Open Science awards for FOSS in France: 2022 and 2023

First edition, 2022 prize



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE
Liberté Égalité Fraternité

Accueil > Recherche > Science ouverte

Publié le 05.02.2022

Sommaire

- The Coq proof assistant : lauréat de la catégorie Scientifique et technique
- Scikit-learn : lauréat de la catégorie Communauté
- Faust : lauréat de la catégorie Documentation
- Gammapy : prix du jury
- Jury

Remise des prix science ouverte du logiciel libre de la recherche

Le ministère de l'Enseignement supérieur, de la Recherche et de l'Innovation remet pour la première année les Prix science ouverte du logiciel libre de la recherche. Dix logiciels mis au point par des équipes françaises sont récompensés pour leur contribution à l'avancée de la connaissance scientifique.

- 129 projects
- 4 awards
- 6 accessit
- first edition
- Coq proof assistant
- Scikit-Learn ML/AI
- Faust music
- Gammapy astronomy

Second edition, 2023 prize



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE
Liberté Égalité Fraternité

BUREAU OFFICIEL VOUS ÊTES

Accueil > Recherche > Science ouverte

Publié le 29.11.2023

Sommaire

- PPyroGOLIN : espoir de la catégorie « Scientifique et technique »
- Smilal : lauréat de la catégorie « Scientifique et technique »
- NeiroCapture : espoir de la catégorie « Communauté »
- OCaml : lauréat de la catégorie « Communauté »
- KeOps : espoir de la catégorie « Documentation »
- Brian : lauréat de la catégorie « Documentation »
- Fink : espoir de la catégorie « Coup de cœur » du jury
- Hybris : lauréat de la catégorie « Coup de cœur » du jury
- Jury

Remise des prix science ouverte du logiciel libre de la recherche 2023

Le ministère de l'Enseignement supérieur et de la Recherche remet pour la deuxième édition les Prix science ouverte du logiciel libre de la recherche. Huit logiciels développés par des équipes françaises sont récompensés pour leur contribution à l'avancée de la connaissance scientifique ou pour le caractère prometteur de leurs travaux.



- 66 projects
- 4 awards
- 4 "espoirs"
- now runs annually

Blueprint and data from the first edition (2021-2022)

Blueprints and analysis available

Open Research Europe

Browse Gateways & Collections How to Publish About Resource Hub

99 Views 12 Downloads 0 Citations

OPEN LETTER

Establishing a national research software award

[version 1; peer review: 2 approved]

Isabelle Blanc Catala, Roberto Di Cosmo, Mathieu Graud, Daniel Le Berre

Violaine Louvet, Sophie Renaudin,

College of experts for source code and software Committee for Open Science

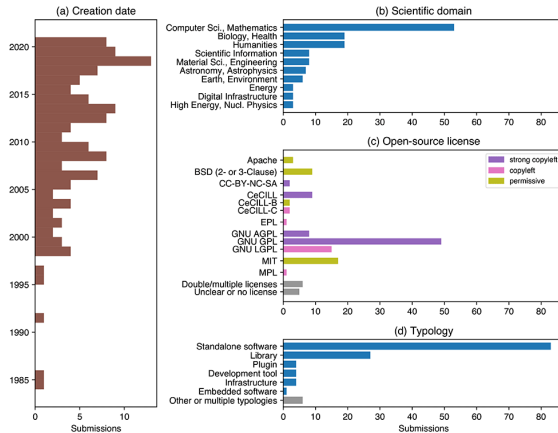


- goals, design decisions
- challenges and solutions
- lessons learned
- detailed data

Emulation is working

Australia: [here](#) and [here](#); Germany: [Helmholtz](#); European Commission: [a CSA](#); ...

A look at the data



National survey of research software (2023)

Context

Article 163 of Law No. 2022-217 of February 21, 2022 required a report on the production and impact of software resulting from research performed in publicly funded entities (universities, research organisations, etc.)

Process and selected results

MINISTÈRE
DE L'ENSEIGNEMENT
SUPÉRIEUR
ET DE LA RECHERCHE
Collège
de la Recherche
Publique



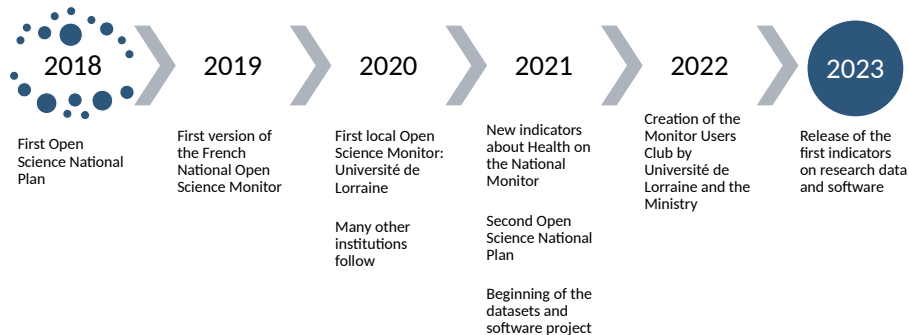
Open survey (1331 detailed answers), and in depth exchanges with tech transfer offices

- 50% of software is older than 9 years
- 36% has more than 100 users
- 62% impact outside academia
- majority is FOSS, 10% proprietary
- 23% undergoes tech transfer

all details (in french) at



A LITTLE BIT OF CONTEXT IN FRANCE...



Credits: Laetitia Bracco and the BSO team

MINING FULL-TEXTS TO DETECT MENTIONS TO DATASETS AND SOFTWARE

- **Innovative approach** based upon the use and development of machine learning tools
 - GROBID: full-text structuring
 - Softcite: **software mention detection**
 - DataStet: **data set mention detection**
- Automatic characterisation of mentions: **usage / production or creation / sharing**
- Another challenge: **downloading massive amounts of full-texts**



science-miner

Alignments were carried out by [ClustalW](#) with default parameters (Thompson et al., 1994). The phylogenetic tree for the *SIDERE2* gene was built using the software program [MEGA4](#) based on protein sequences. The phylogenetic tree was set up with the distance matrix using the Neighbor-Joining (NJ) method with 1000 bootstrap replications. Secondary structure prediction of the *SIDERE2* protein was performed using the program [PSIPRED](#) (Jones, 1999). The *ab initio* structure prediction of the protein was done with the help of [I-TASSER](#) (Zhang, 2009). Automated homology model building of the DNA-binding domain was performed using the protein structure modelling program [MODELLER](#) which models protein tertiary structure by satisfying restraints. The input for [MODELLER](#) consisted of the aligned sequences of Igce and the *SIDERE2* coding file that gives all the necessary commands to the [MODELLER](#) to produce a homology model of the target on the basis of its alignment with the template. Energy minimization was performed by the steepest descent followed by the conjugate gradient method using a 20 Å non-bonded cut-off and a constant dielectric of 1.0. Evaluation of the predicted model involved analysis of the geometry and the stereochemistry of the model. The reliability of the model structure was tested using the [ENERGY](#) commands of [SCOPED](#) (Liu and Huxford, 1992). The modelled structures were also validated using the program [PROSA](#) (Wiederstein and Sippl, 2007).

Southern blot analysis
Genomic DNA of foxglove was extracted from leaves using the cetyltrimethylammonium bromide (CTAB) method (Saghai-Maroof et al., 1984), digested with *PvuII* and *HindIII* (New England Biolabs), fractionated in a 1.0% agarose gel and blotted on a Hybond N⁺ membrane (Amersham). The blots were hybridized to a 705 bp *SIDERE2* probe radioactively labeled with [γ -³²P] dCTP using a High Prime DNA labeling kit (Roche, USA). Hybridization was carried out in 0.5 M sodium phosphate (pH 7.2), 7% SDS, and 1 mM EDTA.

Subcellular localization of the *SIDERE2* protein
The *SIDERE2* gene was fused to the 5' and 3' ends of the green fluorescent protein (GFP) reporter gene using the pCambia 1302 plant expression vector without a stop codon between the *NcoI* and *SpeI* sites. Recombinant DNA constructs encoding the *SIDERE2*-GFP fusion protein downstream of the cauliflower mosaic virus (CaMV) 35S promoter were introduced into onion epidermal cells by gold particle bombardment using the PDS-1000 system (Bio-Rad) at 1100 psi helium pressure. Onion cells were also transiently transformed with the pCambia 1302-GFP vector as a control. Transformed cells were placed on MS solid medium at 22 °C and incubated for ~48 h before being examined. The subcellular localization of GFP fusion proteins was visualized with a confocal microscope (TCS SP2, Leica).

I-TASSER

Type: software

Raw name: I-TASSER

References:

(Zhang, 2008) Zhang (2009)

authors

Yang Zhang

title

I-TASSER: Fully automated protein structure prediction in CASP8

date

2009

journal

Proteins: Structure, Function, and Bioinformatics

volume

77

issue

59

first page

100

last page

113

ISSN

0887-3585

DOI

10.1002/prot.22588

PMC ID

PMC2782770

PMID

19788687

Open

http://neuroepmc.org/articles/pmc27827

Access

pdf-render

publisher

Wiley

I-TASSER (Iterative Threading ASSEMBLY Refinement) is a bioinformatics method for predicting three-dimensional structure model of protein molecules from amino acid sequences. It detects structure templates from the Protein Data Bank by a technique called



Credits: Laetitia Bracco and the BSO team
Uses improved version of SoftCite w.r.t. [the CZI 2022 study in biomedicine](#)

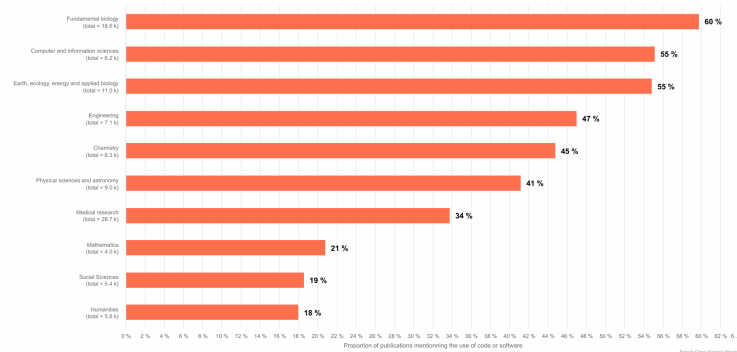
... gives a precious view, per discipline

Software Use

Proportion of publications in France that mention the use of code or software by discipline

Sort by:

☐ Highest volume ☒ Highest use rate



Software is used massively across all disciplines/



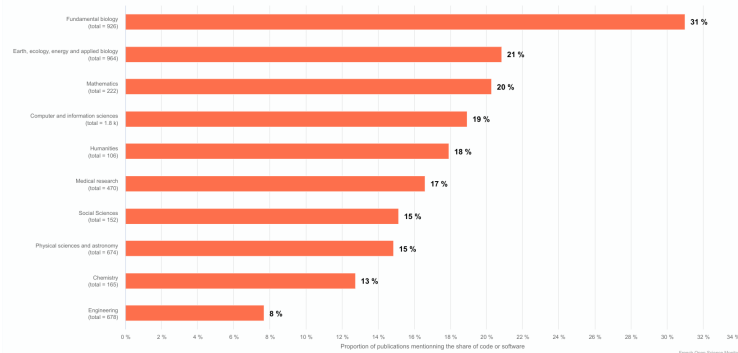
... gives a precious view, per discipline, cont'd

Software Sharing

Proportion of publications in France that mention code or software sharing by discipline

Sort by:

☐ Highest volume ☒ Highest sharing rate



Over 20% of articles mentioning software creation actually **share it**



- 1 Introduction
- 2 Software and Source Code
- 3 France is leading the way
- 4 Addressing the ARCD needs**
- 5 Adoption indicators and call to action

What is at stake

Archive

Research software artifacts must be properly **archived**
make sure we can *retrieve* them (*reproducibility*)

Reference

Research software artifacts must be properly **referenced**
make sure we can *identify* them (*reproducibility*)

Describe

Research software artifacts must be properly **described**
make it easy to *discover* and *reuse* them (*visibility*)

Cite/Credit

Research software artifacts must be properly **cited** (*not the same as referenced!*)
to give *credit* to authors (*evaluation!*)

Software Heritage: *one* software archive, a *shared* infrastructure ...

One infrastructure
open and shared



Inria



The largest archive ever built



Diamond sponsors



Platinum sponsors



Gold sponsors



Silver sponsors

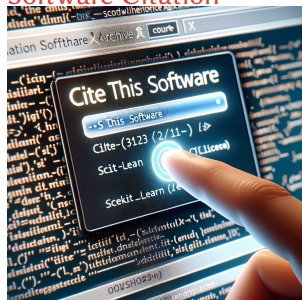


Bronze sponsors



Breaking news: seamless software citation

Software Citation



DALL-E's view of software citation

- SWH is uniquely positioned:
 - platform agnostic: works with all forges, not just GitHub!
 - *only place* to provide the right SWHID for a visited (version of a) project
- in production since January 2025

- 1 Introduction
- 2 Software and Source Code
- 3 France is leading the way
- 4 Addressing the ARCD needs
- 5 Adoption indicators and call to action

A few adoption indicators



Policy



- [Recommendations in ANR 2023 guidelines \(p. 17\)](#)
- HAL+SWH in [the Open Science software booklet](#)

Projects



FAIRCORE4EOSC
Core Components Supporting a FAIR EOSC

The CodeMeta Project



FAIR-IMPACT
Expanding FAIR solutions across EOSC

Users and collaborations

What are they “referencing”?

source	n	percentage
Not available	2868	46.22
GitHub	1151	18.55
software heritage	387	6.24
zenodo	142	2.29
r package	70	1.13
cran	56	0.90
r package version	54	0.87
gitlab	35	0.56

Graphics Replicability Stamp Initiative



b/Surf: Interactive Bézier Splines on Surface Meshes

Claudio Mancinelli, Giacomo Nazzaro, Fabio Pellacini, Enrico Puppo
IEEE Transactions on Visualization and Computer Graphics (TVCG)



Repository



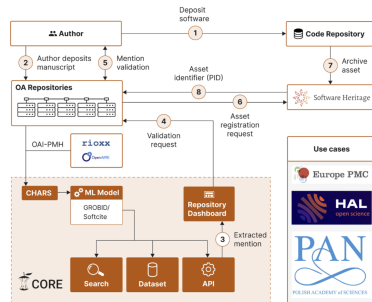
We are going global

Institutional: OSMF



Effort under **UNESCO and France impulse** to build an "open science monitor framework" compatible monitors across countries

Infrastructure: SOFair



Effort to identify software mentions in **all the open access literature**, and add links to the **Software Heritage archive**



Call to action: push adoption of best practices for ARDC

Archive and reference

All **source code** used in research (*yes, even small scripts!*)

for reproducibility

- save in Software Heritage
- add **SWHID** in articles

See [detailed HOWTO online](#)



Describe and Cite/Credit

For **software you want to put forward** (*mention in your CV, reports, etc., get citations and credit for it*), do the following **extra steps**:

video tutorials

- add **codemeta.json** (see the [codemeta generator](#))
- reference in HAL (*french partners*, see [online HAL documentation](#))
- cite using the [biblatex-software](#) package (in CTAN and TeXLive)



train students and colleagues

engage journals, conferences, learned societies

libraries join the ALIG interest group



Call to action: policy making

A working agenda

- **support** open source research software
 - clear **policy** and institutional home (see OSPOs in the US)
 - common **knowledge base** for technology transfer
 - technical and financial **sustainability**
 - modern, efficient, scalable, maintained **collaboration infrastructures**
- establish *intelligent and effective* **incentives** (mind Goodhart's law)
 - count *quality software contributions* in careers
 - avoid *purely numerical indicators*, keep the human in the loop
 - respect software complexity, *dont treat it as data*
- **avoid balkanisation**, support mutualised common infrastructures
 - build on common, shared, open, non profit infrastructures, like **Software Heritage**
 - acknowledge the **predominant human component** of digital infrastructures
 - *recurrent funding* of the cost, for proper *evaluation of the service*
- **lead, dont follow** the international conversation

let's work together to drive the wave

References

-  UNESCO, *Draft recommendations on Open Science* 2021, ([online](#))
-  French Ministry of Research, *Second National Plan for Open Science* 2021, ([online](#))
-  EOSC SIRS Task Force, *Scholarly Infrastructures for Research Software* 2020, Publications office of the European Commission, ([10.2777/28598](#))
-  R. Di Cosmo, *Archiving and Referencing Source Code with Software Heritage* International Conference on Mathematical Software 2020 ([10.1007/978-3-030-52200-1_36](#))
-  J.F. Abramatic, R. Di Cosmo, S. Zacchiroli, *Building the Universal Archive of Source Code* CACM, October 2018 ([10.1145/3183558](#))