Software Heritage: a new infrastructure for Open Science

policy and implementation

Roberto Di Cosmo
IFIP 2022 General Assembly

Director, Software Heritage
Inria and Université de Paris Cité

September 19th 2022

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

Outline

Short Bio: Roberto Di Cosmo

Computer Science professor in Paris, now working at INRIA

- *30+ years* of research (Theor. CS, Programming, Software Engineering, Erdos #: 3)
- *20+ years* of Free and Open Source Software
- *10+ 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

Outline

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 Commissionneer 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."

Two well known pillars of Open Science

Open Access (long, painful, unfinished story)

19XX's compulsory exclusive copyright transfer to publishers
(notable exceptions: US federal agencies and UK Crown Copyright)

1990's Internet, Web and ArXiv break the marriage of convenience of researchers with publishers

2010's reactions (SciHub, 2011; Plan S, 2018) and transformations (not so easy)

TL;DR: see my viewpoint in 2005 and the SIGPLAN blog in 2020

Open Data (less painful, but still unfinished story)

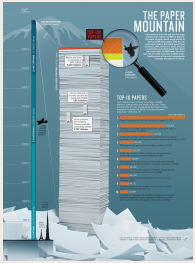
- 2006 (and 2021): OECD recommendation on publicly funded research data
- 2016 and later: FAIR terminology (*focus on metadata, sort of forgets open…*)

Breaking news, August 25, 2022: US OSTP memo to federal agencies

zero embargo on public access to publications and data

Software: the third pillar of Open Science

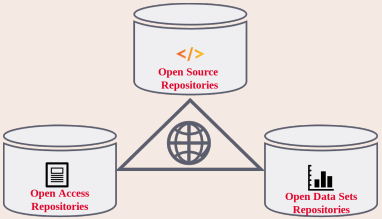
Software powers modern research



[...] software [...] essential in their fields.
Top 100 papers (Nature, 2014)

Sometimes, if you dont have the software, you dont have the data
Christine Borgman, Paris, 2018

A key pillar: software (source code)



The links in the picture 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*

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)

```
P63SPOT3        CA      BIT6            # IS THE LR ANTENNA IN POSITION 1 YET
                EXTEND
                RAND    CHAN33
                EXTEND
                BZF     P63SPOT4        # BRANCH IF ANTENNA ALREADY IN POSITION 1

                CAF     CODE500         # ASTRONAUT:    PLEASE CRANK THE
                TC      BANKCALL        #               SILLY THING AROUND
                CADR    GOPERF1
                TCF     GOTOPOOH        # TERMINATE
                TCF     P63SPOT3        # PROCEED       SEE IF HE'S LYING

P63SPOT4        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."

Outline

International highlights

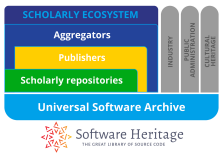
Paris Call on Software Source code (2019, UNESCO)

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"*

UNESCO recommendations for Open Science, 2018-2021

"The source code must be included in the software release and [...] the license must allow modifications, derivative works and sharing [...]"
"Open science infrastructures should be [...] essentially not-for-profit and long-term"

EOSC SIRS report: Software Source Code and Open Science, 2020



- connect scholarly ecosystem via Software Heritage
- use open non profit infrastructures
- open source first: *"all research software should be made available under an Open Source license by default"*

French National plan for Open Science, 2021-2024

MINISTÈRE
DE L'ENSEIGNEMENT
SUPÉRIEUR,
DE LA RECHERCHE
ET DE L'INNOVATION

Liberté
Égalité
Fraternité

SECOND FRENCH PLAN FOR OPEN SCIENCE

Generalising open science in France 2021-2024

1

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

2

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

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

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

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

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

3

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.

4

Outline

A plurality of needs

Researchers

- archive and reference software used in articles
- find useful software
- get credit for developed software
- verify, reproduce, improve results

Laboratories/teams

- track software contributions
- produce reports
- maintain web page

Research Organization

know its software assets

- technology transfer
- impact metrics
- funding strategy
- career evaluation

What is at stake

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, ...)
- Opening up towards a community (documentation, organization, communication)

Need training, best practices

Beyond ARDC

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

let's focus on infrastructures for ARDC

Forges are *not* archives!

2015: the first big bad news

Google Code and Gitorious.org shutdown: ~1M endangered repositories

- broken links in the web of knowledge (my papers too)

Big bad news keep coming in

- summer 2019: BitBucket announces Mercurial VCS sunset
- july 2020: BitBucket erases *250.000+* repositories (including research software)
- summer 2022: GitLab.com considers erasing all projects that are inactive for a year

In Academia too!

- 2021: Inria's old gforge is unplugged… breaks the Opam build chain for OCaml

We need a universal archive of software source code: now we have one!

Outline

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Collect, preserve and share *all* software source code

Preserving our heritage, enabling better software and better science for all

Reference catalog

find and reference all software source code

Universal archive

preserve all software source code

Research infrastructure

enable analysis of all software source code

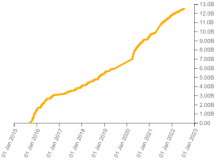
The largest software archive, a shared infrastructure

Cultural Heritage | **Industry** | **Research** | **Public Administration**

Software Heritage

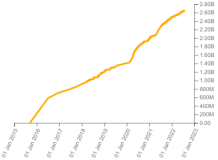
Source files

12,538,666,608



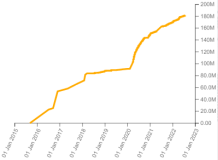
Commits

2,654,066,174



Projects

181,249,577



Directories

10,342,140,231

Authors

48,778,458

Releases

33,580,610

An international, non profit initiative built for the long term

Sharing the vision

United Nations
Educational, Scientific and
Cultural Organization

And many more ...
www.softwareheritage.org/support/testimonials

Donors, members, sponsors

Inria
INVENTEURS DU MONDE NUMÉRIQUE

Diamond sponsor

Platinum sponsors

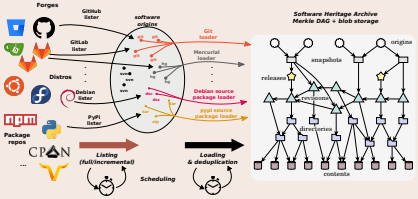
Gold sponsors

Silver sponsors

Bronze sponsors

Addressing the four ARDC needs (see ICMS 2020 for details)

Archive (12B+ files, 170M+ projects)



- save.softwareheritage.org
- deposit.softwareheritage.org

Describe

- *Intrinsic metadata* from source code
- Contributed the Codemeta generator

Reference (20 billion SWHIDs)

Intrinsic, decentralised, cryptographically strong identifiers, SWHIDs

Now supported in SPDX 2.2, Wikidata etc.

Cite/Credit

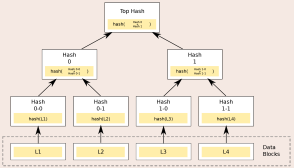
- Contributed *software citation* style biblatex-software, v 1.2-2 now on CTAN

A revolutionary infrastructure

The *graph* of Software Development

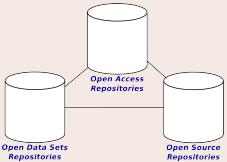
All software development in a single graph …

The *blockchain* of Software Development



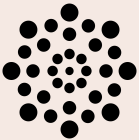
… a single Merkle graph!

A *pillar* of Open Science



Reference archive of Research Software

Reference platform for *Big Code*



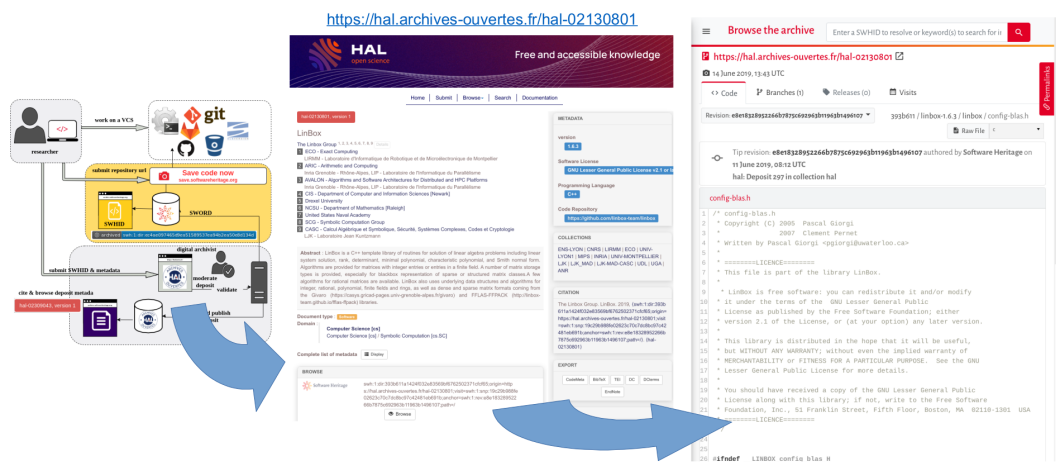
A single, uniform data structure

Outline

A walkthrough

- Browse (e.g. Apollo 11, and your work may be already there !)
- Trigger archival, use the updateswh browser extension (GitHub action available too)
- Get and use SWHIDs (full specification available online)
- Cite software using the biblatex-software package from CTAN
- Example in a journal: an article from IPOL
- Example with Parmap: devel on Github, archive in SWH, curated deposit in HAL
- Extracting all the software products for Inria, for CNRS, for CNES, for LIRMM or for Rémi Gribonval using HalTools
- Curated deposit in SWH via HAL, see for example: LinBox, SLALOM, Givaro, NS2DDV, SumGra, Coq proof, ...
- Example use in research articles:
  - compare Fig. 1 and conclusions in the 2012 version and the updated version
  - SWHID in a replication experiment

Overview of the Software Heritage / HAL synergy



FAIRCORE4EOSC: Dagstuhl - SWH to implement same as the HAL example above

Growing adoption of SWH in Academia (selection)

HAL software curated deposit workflow

Curated Archiving of Research Software Artifacts
International Journal of Digical Curation, 2020

Reference archive for swmath.org

swMATH
an information service for mathematical software

See *code* links, e.g.
SemiPar package

IPOL (image processing)

- archive (deposit)
- reference
- BibLaTeX

eLife (life sciences)

- archive (save code now)
- reference

JTCAM (mechanics)

- instructions for authors
- biblatex-software in journal LaTeX class

Policy: France

NATIONAL PLAN FOR OPEN SCIENCE

National Plan for Open Science and Research Infrastructures

Policy: Europe

EOSC SIRS report

- SWHIDs
- archive

Guidelines

- summary
- ICMS 2020

Outline

Call to action: best practices for ARDC are available… today!

Archiving and referencing

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

- archive and reference in Software Heritage (see detailed HOWTO)

Describing and Citing/Crediting

For software one wants to put forward, add these extra steps:

- add codemeta.json with description (see the codemeta generator)
- (french partners) reference in the HAL portal (see online HAL documentation)
- cite software using the biblatex-software package (in CTAN and TeXLive)

We can (and must)

- train students and colleagues
- engage journals, conferences, learned societies

Policy remarks on the road ahead

Infrastructures for Software: avoid balkanisation, mutualise cost

- build on common, shared, open, non profit infrastructures
- join Software Heritage

development member/sponsor, mirror, contributor
adoption ambassador, learned societies, policy
research address the many scientific challenges

Walking the talk in Europe

ongoing full workpackage in FAIRCORE4EOSC interconnects infrastructures with Software Heritage

open now CHIST-ERA joint ORD call ~~ deadline: 14/12/2022

Belgium, Czech Republic, France, Lithuania, Luxembourg, Poland, Slovakia, Switzerland, Turkey

"Processes and tools to describe, share, reference and archive software [...] that leverage existing initiatives, such as Software Heritage"

Policy remarks on the road ahead, cont'd

Avoid proprietarization of public research result

set the default to open *for software too*

- *publicly funded research software should be open source*, exceptions must be justified

Establish intelligent incentives

> *When a measure becomes a target, it stops being a good measure.*
>
> *Goodhart's law*

In hiring, funding and career evaluation

- avoid *purely numerical* indicators
- count *quality* software contributions in careers (all aspects!)
- keep *the human* in the loop

The floor is yours

IFIP has an important role to play, let's work together!

Questions?

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)