Archiving and referencing software source code

towards a universal infrastructure

Roberto Di Cosmo

October 28th, 2019

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

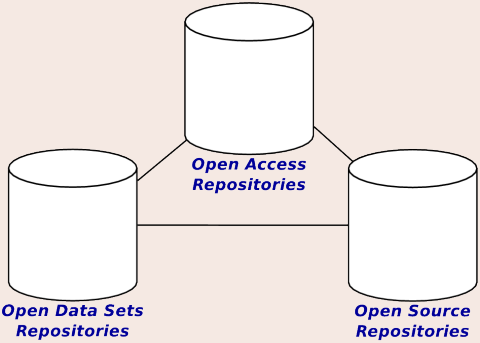
2015 *Software Heritage* at INRIA

2018 *National Committee for Open Science*, France

Software is everywhere

At the heart of our society

A pillar of modern research



Software embodies our collective Knowledge and Cultural Heritage

Pressure to make research software source code available is raising

Why

Necessary to

- *reproduce* and verify,
- *modify* and *evolve*, building new experiments from old ones

When and where

- debate started end of first 2000 decade (biology, statistics, medicine, etc.)
- growing in Computer Science since the ESEC/FSE 2011 Artifact Evaluation context (winner: Vouillon and Di Cosmo)

Outline

Source code is *special*

Executable and *human readable* knowledge — copyright law

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

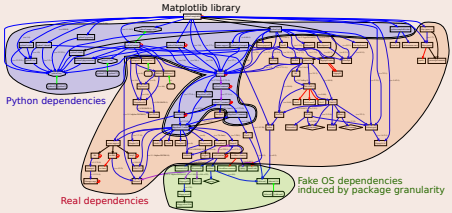
Harold Abelson

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 external dependencies*
  - easy to break, difficult to maintain
- sophisticated *developer communities*



What we need to address in increasing order of difficulty

Archival

Research software artifacts must be properly archived

make it sure we can *retrieve* them (*reproducibility*)

Identification

Research software artifacts must be properly referenced

make it sure we can *identify* them (*reproducibility*)

Metadata

Research software artifacts must be properly described

make it easy to *discover* them (*visibility*)

Citation

Research software artifacts must be properly cited *(not the same as referenced!)*

to give *credit* to authors (*evaluation*!)

Outline

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Collect, preserve and share the *source code* of *all the software*

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

Reference catalog

find and reference all the source code

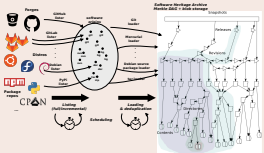
Universal archive

preserve all the source code

Mutualised infrastructure

industry, research, culture

The largest software source code archive *ever*



20 billions intrinsic identifiers for reproducibility

Compatible with git see `bit.ly/swhpidpaper`

Reference archive

See the work done at *swmath.org*

SWH IDs now a standard for Wikidata

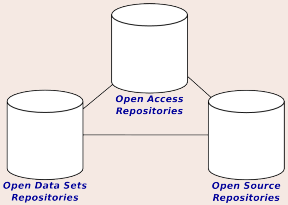
See `https://www.wikidata.org/wiki/Property:P6138`

Policy

Now part of the *French National Plan for Open Science*

Software Heritage for Open Science

A revolutionary infrastructure



- universal archive of all source code
  - we archive *all* software: both research and non research
  - we *proactively collect software* in a systematic way
- intrinsic identifiers for reproducibility without IPC!
  - reference software artefacts *without any third party*
  - cryptographically strong, compatible with git hashes
- also save code now and curated deposit (e.g. via HAL)

Guidelines are now available

- blog with overview: `http://bit.ly/blogsaveres`
- full details: `http://bit.ly/swhsaveguide`

Outline

Challenges still in front of us

Handle various granularities

- ☒ Exact status of the source code for reproducibility, e.g.

"you can find at swh:1:cnt:cdf19c4487c43c76f3612557d4dc61f9131790a4;lines=146-187 the core algorithm used in this article"

- ☐ (Major) release *"This functionality is available in OCaml version 4"*
- ☐ Project *"Inria has created OCaml and Scikit-Learn"*.

Accomodate Software Complexity in Metadata and Citation

Structure	monolithic/composite; self-contained/external dependencies
Lifetime	one-shot/long term
Authorship	complex set of roles
Authority	institutions/organizations/communities/single person

needs *proper human curation*

Conclusion

Software for Open Science: we're halfway through

- Software Heritage solves *archival and identification*
- for *metadata* and *citation* a lot more work is needed

Thank you!

Jean-François Abramatic, Roberto Di Cosmo, Stefano Zacchiroli
Building the Universal Archive of Source Code
Communications of the ACM, October 2018

Roberto Di Cosmo, Morane Gruenpeter, Stefano Zacchiroli
Identifiers for Digital Objects: the Case of Software Source Code Preservation
iPRES 2018: Intl. Conf. on Digital Preservation

Pierre Alliez, Roberto Di Cosmo, Benjamin Guedj, Alain Girault, Mohand-Said Hacid, Arnaud Legrand, Nicolas P. Rougier.
Attributing and Referencing (Research) Software: Best Practices and Outlook from Inria.
https://hal.archives-ouvertes.fr/hal-02135891, May 2019.