Archiving and Referencing all the source code
working together to make software count

Roberto Di Cosmo
Director, Software Heritage
Open Access Week

November 5th, 2020
Outline

1. Software Source Code is knowledge
2. Software Heritage
3. Demo time!
4. The way forward
5. Conclusion
Software source code: *human readable and executable 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)

```plaintext
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 CODE580 # ASTRONAUT: PLEASE CRANK THE
TC BANKCALL # SILLY THING AROUND
CADR GOPERF1
TCF GOTOP00M # 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)

```plaintext
float Q_rsqrt(float number)
{
    long i;
    float x2, y;
    const float threehalves = 1.5F;
    x2 = number * 0.5F;
    y = number;
    i = * (long *) &y; // evil floating point bit level hacking
    i = 0x5f3759dfL + (1L >> 1); // what the fuck?
    y = * (float *) &i;
    y = y * (threehalves - (x2 * y * y)); // 1st iteration
    // y = y * (threehalves - (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.*”

@rdicosmo Roberto Di Cosmo @swheritage
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*

**Precious, endangered executable and human readable knowledge**
key people *passing away*, platforms (GoogleCode, Gitorious, etc.) closing down …
no organised effort to catalog and archive it
Source code is *special*, cont’d

**Versioning, granularity**

**Project** “Inria created OCaml and Scikit-learn”

**Release** “2D Voronoi Diagrams were introduced in CGAL 3.1.0”

**Precise state of a project** “This result was produced using commit 0064fbd…”

**Code fragment** “The core algorithm is in lines 101 to 143 of the file parmap.ml contained in the precise state of the project corresponding to commit 0064fbd…”

**Authors can have multiple roles:**

- Architecture, Management, Development, Documentation, Testing, …
Software is everywhere in modern research

[...] software [...] essential in their fields.

Top 100 papers (Nature, 2014)

Sometimes, if you don’t have the software, you don’t have the data

Christine Borgman, Paris, 2018

Nota bene

The links in the picture are essential
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 for: technology transfer, impact metrics, strategy

@rdicosmo Roberto Di Cosmo @swheritage
In increasing order of difficulty:

**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!)

Need infrastructures *designed* for software:
now we have one!
Outline

1. Software Source Code is knowledge
2. Software Heritage
3. Demo time!
4. The way forward
5. Conclusion
Software Heritage, in a nutshell

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
An international, non profit initiative built for the long term

Sharing the vision

United Nations Educational, Scientific and Cultural Organization

Donors, members, sponsors

Platinum sponsors

Microsoft

intel

societe generale

Gold sponsors

Huawei

Sorbonne Université

Silver sponsors

GitHub

Google

SIF

Bronze sponsors

dans

fossid

Creative Commons

And many more ...

www.softwareheritage.org/support/testimonials

www.softwareheritage.org/support/testimonials
Addressing the four ARDC needs (see ICMS 2020 for details)

Archive (8B+ files, 140M+ projects)

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

Reference (20 billion SWHIDs)

Intrinsic, decentralised, cryptographically strong identifiers, SWHIDs

Now supported in SPDX 2.2, Wikidata etc.

Describe

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

Cite/Credit

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

@rdicosmo Roberto Di Cosmo @swheritage

Archiving and Referencing source code CC-BY 4.0 November 5th 2020 10 / 15
A walkthrough

**Archive**
- **Trigger archival** of your preferred software in a breeze
- **curated deposit in SWH via HAL**, see for example: LinBox, SLALOM, Givaro, NS2DDV, SumGra, Coq proof, …
- rescue landmark legacy software, see the SWHAP process with UNESCO

**Reference**
- Browse the archive
- Get and use SWHIDs (full specification available online)
- cite software using the biblatex-software style

**Cite/Credit**
- Example use in a research article: compare Fig. 1 and conclusions
  - in the 2012 version
  - in the updated version using SWHIDs and Software Heritage
1. Software Source Code is knowledge
2. Software Heritage
3. Demo time!
4. The way forward
5. Conclusion
Adoption is coming

**HAL software curated deposit workflow**

*Curated Archiving of Research Software Artifacts*
International Journal of Digital Curation, 2020

**Reference archive for swmath.org**

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

**Image Processing On Line (IPOL)**

- archives
- reference
- cite: see BibLaTeX example

**JTCAM (Theor. Comp. and Appl. Mech)**

- instructions for authors recommend archival in Software Heritage
- biblatex-software in journal \LaTeX{} class

**Policy**

now officially in the **French National Plan for Open Science**

**Self archival guidelines**

- online summary
- full ICMS 2020 paper

---

@rdicosmo Roberto Di Cosmo @swheritage

Archiving and Referencing source code CC-BY 4.0 November 5th 2020 12 / 15
Breaking news: saving 250,000 endangered repositories

Bitbucket phase out of Mercurial VCS

- summer 2019: official announcement
- fall 2019: Software Heritage teams up with Octobus (funded by NLNet, thanks!)
- july 2020: 250,000 repositories unplugged
- august 2020: bitbucket-archive.softwareheritage.org is live

... preserving the web of knowledge

Bottomline

*explicit deposit* is important, …

… and we must promote it…

… but will never be enough.

*(think also of all software dependencies!)*

---

@rdicosmo Roberto Di Cosmo @swheritage

Archiving and Referencing source code  CC-BY 4.0  November 5th 2020
Breaking news: a roadmap for software in the EOSC

Infrastructures in the architecture

universal software archive  *Software Heritage* connects with the global software ecosystem

scholarly repositories  *HAL, Zenodo, …*

publishers  *Dagstuhl, eLife, IPOL, …*

aggregators  *OpenAire, ScanR, swMath, …*

Towards interconnection and interoperability

metadata standard  proposal to adopt *CodeMeta*

intrinsic identifiers  proposal to adopt *SWHID*

extrinsic identifiers  take into account what exists

**EOSC SIRS TF report:** community review until 10/11/2020
The way forward

Software Heritage

- universal archive of source code
- intrinsic identifiers (SWHIDS)
- non profit, long term, multistakeholder
- infrastructure for Open Science

Your help is needed!

- adopt use SWH in your work
- save relevant source code
- contribute SWH is open source
- advocate spread the word

Roberto Di Cosmo
Archiving and Referencing Source Code with Software Heritage
International Congress on Mathematical Software (ICMS), 2020

Jean-François Abramatic, Roberto Di Cosmo, Stefano Zacchirol
Building the Universal Archive of Source Code, CACM, October 2018 (10.1145/3183558)

Pierre Alliez, Roberto Di Cosmo, Benjamin Guedj, Alain Girault, Mohand-Said Hacid, Arnaud Legrand and Nicolas Rougier
Attributing and referencing (research) software: Best practices and outlook from Inria,
CiSE 2020 (10.1109/MCSE.2019.2949413) (hal-02135891)