Archiving, referencing and attributing research software
towards software as a first class citizen

Roberto Di Cosmo
Inria and Université de Paris

February 13th, 2020
1. Software Source Code: a (forgotten) pillar of Science
2. Meet Software Heritage
3. Archive and reference all the source code
4. Describe and cite research source code
5. The road ahead
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.”

Software source code: a precious part of our heritage

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 GOTOPOOM  # 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 ...
```

Quake III source code (excerpt)

```
float q_sqrt(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 = 0x5f3759df - (i >> 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

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

Roberto Di Cosmo

(CC-BY 4.0) Research Software February 13th, 2020 1 / 23
Source code is a **special** and endangered heritage

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
- sophisticated *developer communities*

**Precious, endangered Executable and human readable knowledge**
- key people are passing away ...
- no organised effort to catalog and archive it
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
The state of the art (in CS!) is far from ideal

ICSE (Zannier, Melrik, Maurer, 2006)
- complete absence of replication studies

ACM TOSEM 2001 to 2006 (C. Ghezzi)
- 60% of all papers have tools: only 20% installable

Collberg’s 2015 study
- 601 mainstream papers: 508 with tools, only 40% installable

Main reasons
- source code (or the right version of it) cannot be found
Where we stand

A wealth of initiatives!

- Policies: ACM Artifact Review and Badging, …
- Working groups: FORCE11, RDA, SPSO, …
- Metrics: Open Science Monitor (Elsevier!), …
- Repositories: FigShare, Zenodo, …

but …

Lack of recognition

not (yet) a first class citizen

- in the EOSC plan
- in the scholarly works

Lack of proper guidance on how to

- archive and reference software
- choose a license
- cite a software project
What is at stake

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!*)
## A plurality of needs

### Researcher
- archive and reference sw used in articles
- get credit for the software they develop
- verify/reproduce/improve results

### Laboratory/team
- track software contributions
- produce up-to date report / web page

### University/Research Organization
- central view of research software assets
- tech transfer
- impact metrics
Outline

1. Software Source Code: a (forgotten) pillar of Science

2. Meet Software Heritage

3. Archive and reference *all* the source code

4. Describe and cite *research* source code

5. The road ahead
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

Research infrastructure
enable analysis of all the source code
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

Platinum sponsors

Microsoft

Société Générale

Huawei

Gold sponsor

openinventionnetwork

Université de Paris

Silver sponsors

CAST

GitHub

Google

vmware

Bronze sponsors

DANS

FOSSID

Nokia Bell Labs

UQÀM

University of Quebec à Montréal

Roberto Di Cosmo

(CC-BY 4.0) Research Software February 13th, 2020
The largest software archive, a shared infrastructure
A peek under the hood

Global development history permanently archived in a unique git-like Merkle DAG

- ~400 TB (uncompressed) blobs, ~20 B nodes, ~280 B edges
Software Heritage for Research and Innovation

Reference platform for *Big Code*

- unique *observatory* of all software development
- *big data, machine learning* paradise: classification, trends, coding patterns, code completion…

First datasets are available!

- full graph of software development (~20Bn nodes, ~200Bn edges) see Pietri, Spinellis, Zacchiroli, MSR 2019
  
  https://dx.doi.org/10.1109/MSR.2019.00030

- MSR 2020 mining competition see https://2020.msrconf.org/track/msr-2020-mining-challenge#Call-for-Papers
Raising awareness about Software Source Code

UNESCO, Inria, Software Heritage invite 40 international experts meet in Paris ...

Their call is published on Feb 2019

It’s an important policy tool, already referenced and used ... yes, you can sign it!

Outline

1. Software Source Code: a (forgotten) pillar of Science
2. Meet Software Heritage
3. Archive and reference *all* the source code
4. Describe and cite *research* source code
5. The road ahead
Software Heritage: 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
  - identify software artefacts without any third party
  - cryptographically strong, compatible with git hashes

Full guidelines available!  https://www.softwareheritage.org/save-and-reference-research-software/

Save code now … in just a few clicks

Demo

My 2012 Parmap paper before and after; other links: Apollo 11 (and blog), Quake III Arena
The SWH-ID schema

```
swh:1:cnt:41ddb23118f92d7218099a5e7a990cf58f1d07fa
```

- `schema_version`:
- `object_id`:
- `prefix`:
- `object_type`:
The SWH-ID schema

- **schema_version**: 1
- **object_id**: 41ddb23118f92d7218099a5e7a990cf58f1d07fa

```
swh:1:cnt:41ddb23118f92d7218099a5e7a990cf58f1d07fa
```

- **prefix**: swh
- **object_type**:
  - "snp" - snapshot
  - "rel" - release
  - "rev" - revision
  - "dir" - directory
  - "cnt" - content
The SWH-ID schema

```plaintext
swh:1:cnt:41dab23118f92d7218099a5e7a990cf58f1d07fa

- prefix
- object_type
  - "snp" - snapshot
  - "rel" - release
  - "rev" - revision
  - "dir" - directory
  - "cnt" - content

- object_id
- schema_version

;lines=64-72
;origin=https://github.com/chrislgarry/Apollo-11
```
A worked example

Contents

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of
this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom
to share and change all versions of a program--to make sure it remains
free software for all its users. We, the Free Software Foundation, use
the GNU General Public License for most of our software; it applies also
to any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or changing your program to deny you them.

This General Public License applies to most software on the
Internet today.

SHA1: 8624bcdae55baeef...
SHA256: 8ceb4b9ee5aded...
SHA1 Git: 94a9ed024d385...
Length: 35147
A worked example
A worked example

Directories

- gitignore
- AUTHORS
- LICENSE
- MANIFEST.in
- Makefile
- Makefile.local
- README.db_testing
- README.dev
- bin
- debian
- docs
- requirements.txt
- setup.py
- sql
- swh
- utils

100644 blob c5baade4c44766042186ef858c0fd63d587ebf09 .gitignore
100644 blob 2d6a34af6f52cf3cf6b8c2f7bd0648fbd255e77f AUTHORS
100644 blob 94a9ed824d3859793618152ea559a168bbcb5e2 LICENSE
100644 blob d9b2665a435a43f8a79a84e0867751dfb959c7bb MANIFEST.in
100644 blob 524175c2bad0b35b97f5f7928f2cf5a6d5eaf2eb4 Makefile
100644 blob 5c7e3a5b6bbdb038682ba7793f440492ed9678bb3 Makefile.local
100644 blob 8617980629c2d4e6080404f9a74b985b3e67b README.db_testing
100644 blob 76b299f4c815e0869c414d38d78d7ce08ec51e README.dev
040000 tree ele10ecef948af0b93ab0372af89f12e92618a bin
040000 tree 83e56d8beaf7793c77a45a345c80fcb8af503013 debian
040000 tree a34c9c4ba21390cedc67f9816348d2795557af5 docs
100644 blob f2a6d32c6135a7287bbd76167b01df2ae4f1539 requirements.txt
100755 blob eee147c36caf1bdc2d820da8dc026c5b568180c setup.py
040000 tree 224bb4c1f4c67fca7d160bfdf2d06094e7e1abf3 sql
040000 tree 8631c9cd77bb93168107ab5ba51f40c6300be swh
040000 tree 8bf905b56ba8ed692f1209b2773b474c61d66c1 utils

id: 515f00d44e92c65322aaa9bf3fa097c00dddb9c7d
## Revisions

<table>
<thead>
<tr>
<th>Details</th>
<th>Changes</th>
<th>Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHA: 963634dca6ba5dc37e3ee426ba091092c267f9f6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Author: Nicolas Dandrimont <a href="mailto:nicolas@dandrimont.eu">nicolas@dandrimont.eu</a> (Thu Sep 1 14:26:13 2018)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Committer: Nicolas Dandrimont <a href="mailto:nicolas@dandrimont.eu">nicolas@dandrimont.eu</a> (Thu Sep 1 14:26:13 2018)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subject: provenance.tasks: add the revision -&gt; origin cache task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parent: fc3a8b59ca1df424d860f2c29ab07fee4dc35d10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>provenance.tasks: add the revision -&gt; origin cache task</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
515f00d44e92c65322aaa9bf3fa097c00ddb9c7d
test...storage: properly pipeline origin and cont...
```

```
515f00d44e92c65322aaa9bf3fa097c00ddb9c7d
parent: fc3a8b59ca1df424d860f2c29ab07fee4dc35d10
author Nicolas Dandrimont <nicolas@dandrimont.eu> 1472732773 +0200
committer Nicolas Dandrimont <nicolas@dandrimont.eu> 1472732773 +0200

provenance.tasks: add the revision -> origin cache task

id: 963634dca6ba5dc37e3ee426ba091092c267f9f6
```
A worked example
Releases

object c0c9f16b1e134f593e7567570a1761b156e6eb1d
 type commit
tag v0.0.51
tagger: nicolas dandrimont <nicolas@dandrimont.eu>
Date: Wed Apr 14 13:36:03 2020 +0200
Release swh.storage v0.0.51
  - Add new metadata column to origin_visit
  - Update swh-add-directory script for updated API
-----BEGIN PGP SIGNATURE-----
IQIzBAABCAAdBQIkJxZTNF6uxaWNvG6QGRhmRyaWlbnQuZXUACxqQ7AWLMo2+n
neqprw/aed6505bd5ajwKe+4kWlK1yXqVz5+1k1xXeV1jW/KAwX8xK7xX2kELEdT7uv
ahpZ6Zp3blnQ6acC1irXrS8cth3L2YtrdZeWXXWppkzwWWMAeOd7db8qphwhhBAAdS12
ICBI1zjKcXU6GD93kehKFPwzzXzXo+bs6PsMW3y35Rs67v7Z7aKFMujPpGlyHPY55yo
IGEnWmno7VHy1VymdbLXnZzX7585MaRyaa-A=becqdbuZ7Zzjy+JpUqC8cy493hrn/FL
q5z7muUykyuzH8hGlHjypV+3s6BlnPoO3STI6Z6zjUjpV6p9h73Fpq6t dintZiKcaoz
kJlGtAWYJ86Mev =nKvJeLbr3y+jyWBF3Q5p5a1V8x07An6F1dALCmNMeoKCoKMt
q3jGMax11jgOE7mqFW7qG6sXwppKPhM7hZQLo3nV3GeQ7TrnLUplMc026Hh97AawC
GgK/1phT4zuOLhiO6ryYPy3eV5GVfuDVU39VfQZ92K/fN=5z2w3GMDdRj5LjQ0Mn
RpTTFUSXexUXHOGPqXh5Tytv1p9cpC76U3TSK0K64A2Z1kDmGwXCVPVqYo
U1lih5B5HbMlOyStF8750jU1K7061jFRRUGK0WdoRKOeSKxKWOZGyK3y6rOjU29
q4wir5Q7pqQ2C39OMZAL+HvPAkVysCMejgh6zIr+jEhVuyk=
=KOoP
-----END PGP SIGNATURE-----

id: 85083a5cc14a441c89dea73f5bdf67c3f9c6afdb
A worked example
A worked example

Roberto Di Cosmo

Snapshots

commit 08feb25770109522eb3c3e21691466c53a1dd9158 refs/heads/atime
commit ba544343a4eaf9fe23b46c729cc4f4ce61c67frefs/heads/directory.listing-arrays
commit db92da08192838f569b27fb1c85f6273889c5 refs/heads/ftp
commit c77f7ed5aa6b2227f89f075aa8b3f67de68e088 refs/heads/master
commit 7eca197ff6c6ed2d820447ef54bd1ed9b9a0434631a8f6c2 refs/heads/tmp/tpg-stream-addr
commit 642a265f1510e595b5e5047f3b53ee4f522522e02 refs/heads/tmp/generic-releases
tag 2910f3b1378c77e0965597790f69907c75ff755 refs/tags/v0.1
	72a21991a384e539969dbb5b/f8b0beeb72aee2cdf refs/tags/v0.1.10
	ag 3599eaba68b76b853757670f5a23bbf54f45f5cc refs/tags/v0.1.11

tag 33378245a483ba593a7777b0d6676f30x556 refs/tags/v0.1.12

tag 6574e5275b327cf590311c2bfa936c3b2b3d5 refs/tags/v0.1.13
	5a6325fe68b8b4b561e44280d92a1e1e32f3bd refs/tags/v0.1.14

tag 586febad4580b45f5a8559306743cb3b1c9c77 refs/tags/v0.1.15

tag 8c5885f4998f3633177742bd289f660e5be51c refs/tags/v0.1.16

tag a54244ae3f79fbee33ef2827ee653899abc7d6 refs/tags/v0.1.17

tag 22a27f1590d12222e565554632e166fc4993d9 refs/tags/v0.1.18

tag 699679ad485d4798c8d24aad090ce282536ef47c refs/tags/v0.1.19

tag 32bf5a59fc2a31b3aa6e5f115a5ad382ec75a67 refs/tags/v0.1.20

tag 3147fcb335ec6fc6f492f780e168b1237ebdff2c7 refs/tags/v0.1.20
	215ea50dab1110e82e0b7e76eb4b8073a87908 refs/tags/v0.1.21

tag 3fb18c2872a5d625212124257a1a5d7c9f5fffa1df refs/tags/v0.1.22
	tag 8c0bee80d4437f5c4a262789e460a16ac3c72aba4 refs/tags/v0.1.23

...
Zoom on the trust model for identifiers

Trust model for usual DOIs

Trust model for DOIs with checksums

Trust model for SWH-IDs

Roberto Di Cosmo

(CC-BY 4.0) Research Software

February 13th, 2020
Outline

1. Software Source Code: a (forgotten) pillar of Science
2. Meet Software Heritage
3. Archive and reference *all* the source code
4. Describe and cite *research* source code
5. The road ahead
Many articles/guidelines
- reproducibility
- archival
- credit and evaluation

Most common limitations
- software is ’just data’
- citation = reference = DOIs
- citation produced by automated tools

A few remarkable exceptions
- **ASCL** (since 1999): metadata only, carefully curated
- **geodynamics.org**: source, documentation, metadata
- **swmath.org**: software catalog via articles

Software Citation WG at Inria (since 10/2018)
- leverage a 50 year experience, make recommendations
- read more [https://hal.archives-ouvertes.fr/hal-02135891](https://hal.archives-ouvertes.fr/hal-02135891)
## Why it is not simple

### Software is complex

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>monolithic/composite; self-contained/external dependencies</td>
</tr>
<tr>
<td>Lifetime</td>
<td>one-shot/long term</td>
</tr>
<tr>
<td>Community</td>
<td>one man/one team/distributed community</td>
</tr>
<tr>
<td>Authorship</td>
<td>complex set of roles <em>(more later)</em></td>
</tr>
<tr>
<td>Authority</td>
<td>institutions/organizations/communities/single person</td>
</tr>
</tbody>
</table>

### 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”.
Proposals for the scholarly world

Refined ontology for contributors
- Design, Architecture,
- Coding, Testing, Debugging,
- Documentation, Maintenance, Support,
- Management

see also CRediT, Geodynamics

Reference is distinct from citation
- Reference is for \textit{reproducibility}
- Citation is for \textit{credit}

They must not be conflated

Beware of the numbers game:

\ldots do we really want an \textit{s-index}?

Keep the human in the loop
When \textit{credit} is at stake, automation/crowdsourcing is not enough!

Humans \textit{are needed} to get quality information
First steps with HAL / Software Heritage

How it works, what is special

Generic mechanism:
- SWORD based
- review process
- versioning

Today: deposit .zip or .tar.gz file (guide)
Tomorrow: just provide the SWH id

Deposit/describe research software in HAL
- author: https://hal.archives-ouvertes.fr/hal-01872189
- moderator: https://hal.archives-ouvertes.fr/hal-01876705

Examples
LinBox, SLALOM, Givaro, NS2DDV, SumGra, Coq proof, …
The swmath.org approach

Article based citation

See for example:

- SemiPar on swmath.org
1. Software Source Code: a (forgotten) pillar of Science
2. Meet Software Heritage
3. Archive and reference all the source code
4. Describe and cite research source code
5. The road ahead
We need to care more about research software

You can help make a change

- leverage Software Heritage in conferences, journals, AEC for archival and reference
  https://www.softwareheritage.org/save-and-reference-research-software/
- join the conversation on software citation and software evaluation criteria
- tackle the scientific problems: big code, classification, infrastructure, etc.

Thank you!

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

Roberto Di Cosmo, Morane Gruenpeter, Stefano Zacchiroli
Referencing Source Code Artifacts: a Separate Concern in Software Citation,
CiSE 2020 (10.1109/MCSE.2019.2963148) (hal-02446202)

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)