Garantir un Accès Indépendant au Bien Commun Logiciel

Stefano Zacchiroli

Télécom Paris, Institut Polytechnique de Paris
stefano.zacchiroli@telecom-paris.fr

22 Novembre 2022
Journée Infrastructures pour la Souveraineté Numérique
CNAM, Paris, France

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

(Free) Software is everywhere in society

The Commons and Free/Open Source Software

Definition (Commons)

The commons is the cultural and natural resources accessible to all members of a society, including natural materials such as air, water, and a habitable earth. These resources are held in common, not owned privately.

Definition (Software Commons)

The software commons consists of all computer software which is available at little or no cost and which can be altered and reused with few restrictions. Thus *all open source software and all free software are part of the [software] commons.* [...]

Kranich and Schement (2008); Schweik and English (2012).

But *where* is the software commons distributed from?

- GitHub: 140 M repositories — operated by: Microsoft, USA
- GitLab.com: 4 M — GitLab, Inc., USA
- Bitbucket: 1.9 M — Atlassian, Autralia
- NPM: 1.8 M packages — Microsoft, USA

source: Software Heritage archive; total coverage: 188 M software origins (Nov. 2022)

Software source code is fragile

Like all digital information, FOSS is fragile

- link rot: projects are created, moved around, removed
- business decisions (e.g., Gitorious, Google Code, Bitbucket)
- geopolitics (e.g., embargoes, "intellectual property" regulations, wars)

This matters to *you* (even if you don't know about it). One day it will show up at your door, most likely to break your software supply chain!

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

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

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 and share all software source code

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 and share all software source code

Research infrastructure

enable analysis of all software source code

The largest public source code archive, principled

bit.ly/swhpaper

archive.softwareheritage.org

The largest public source code archive, principled

bit.ly/swhpaper

archive.softwareheritage.org

Technology

- transparency and FOSS
- replicas all the way down

Content (billions!)

- intrinsic identifiers
- facts and provenance

Organization

- non-profit
- multi-stakeholder

An international, non profit initiative built for the long term

Sharing the vision

United Nations Educational, Scientific and Cultural Organization

www.softwareheritage.org/support/testimonials

Donors, members, sponsors

Diamond sponsor

Platinum sponsors

Gold sponsors

Silver sponsors

Bronze sponsors

www.softwareheritage.org/support/sponsors

Not the end of the story

Takeaways

- Continued access to the entire public body of free/open source software is a strategic need for all of society, including both public and for-profit players.
- Software Heritage guarantees today an independent access to the largest publicly accessible share of the software commons.
- The archive is France-based and funded by a diverse set of international public/private and non-profit/for-profit actors.

Discussion

- Depending on a FOSS component is more than needing its code at any given point in time (e.g., upgrades, support, community, etc.). How do we "archive" that?
- Funding: for true independence, we need to depend *less* on non "local" funds. (Hello french IT companies!) But at the same time diversity is *good* for long-term survival.
- Is FOSS all the same, just because it *could* be audited? Or do we care where contributions come from? (Geopolitics again!)

Appendix

Software *source code* is precious human 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."

Software *source code* is precious human 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
```

Software *source code* is precious human 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;
}
```

Software *source code* is precious human 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
```

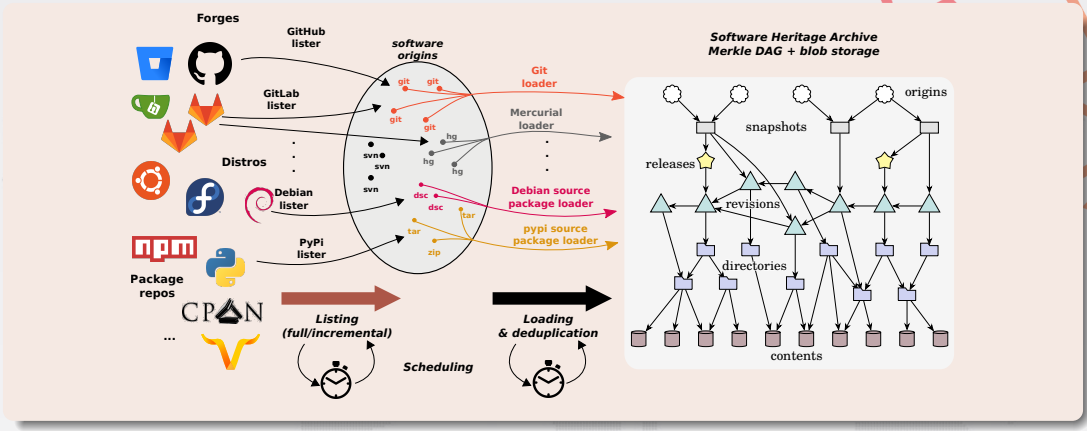
Quake III source code (excerpt)

```
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;
}
```

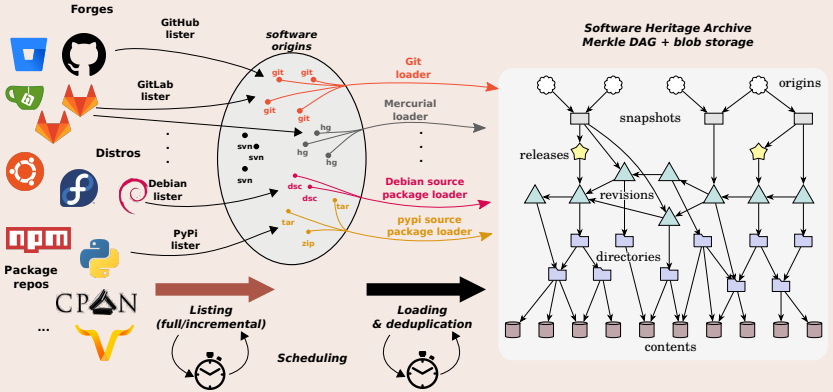
Len Shustek, Computer History Museum 2006

"Source code provides a view into the mind of the designer."

A peek under the hood: a global view on the software commons



A peek under the hood: a global view on the software commons



A global graph linking together fully deduplicated source code artifact (files, commits, directories, releases, etc.) to the places that distribute them (e.g., Git repositories), providing a unified view on the entire *Software Commons*.
(Size: ~30 B nodes, ~300 B edges, ~1 PiB blobs)

Demo time!

- Browse the archive
- Trigger archival of your preferred software in a breeze
- Get and use SWHIDs (full specification available online)
- The Apollo 11 AGC source code example
- Cite software with the biblatex-software style from CTAN
- Example use in a research article: compare Fig. 1 and conclusions
  - in the 2012 version
  - in the updated version using SWHIDs and Software Heritage
- Example in a journal: an article from IPOL
- 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

Academia & policy: growing adoption (selection)

HAL software curated deposit workflow

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

Reference archive for swmath.org

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

Academia & policy: growing adoption (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

Academia & policy: growing adoption (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 and Research Infrastructures

Policy: Europe

EOSC SIRS report

- SWHIDs
- archive

Guidelines

- summary
- ICMS 2020