



Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

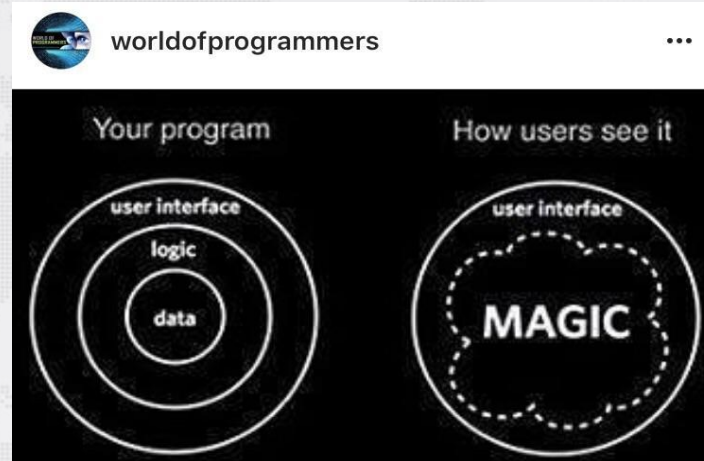
Challenges and opportunities in Academia towards **Open Source** software

Enjeux de **l'ouverture** des logiciels académiques

Morane Gruenpeter
Director of Scholarly Ecosystems
Software Heritage, Inria Foundation



What is software?



Software as a concept

- **project** or entity
- the **community** around the project
- the software **idea** / algorithms / solutions

Software artifacts

- Executables
 - For multiple environments
- **Source code**

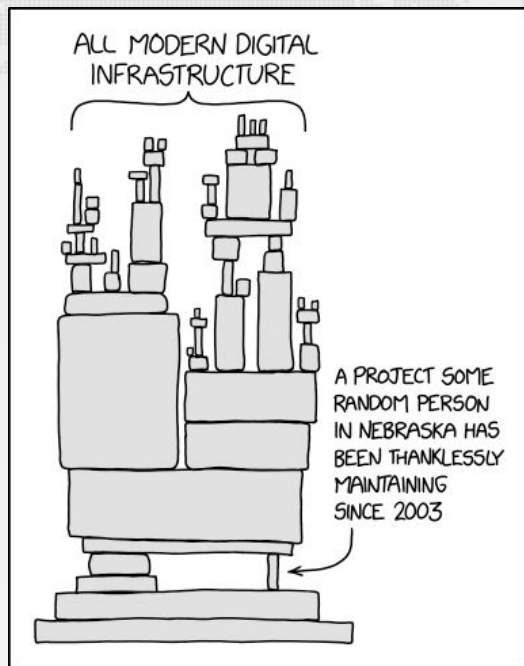
Not a digital artifact

A very large collection of digital artifacts

Source code is...



The invisible backbone of all digital infrastructure



<https://xkcd.com/2347/>

Is #DigitalSovereignty possible without it?

Goals



- ★ **Introduction: why are we here?**
- ★ **Open Science** isn't complete without **Open Source**
- ★ Guided exploration: into the **Software Heritage** archive
- ★ Good practices for **#BetterMetadata**
- ★ The role of libraries: collaborating with **OSP**Os

Why am I here?

1. **Raise awareness** about the universal source code archive.
2. **Share** vision on **Openness** in research and in software engineering
3. **Promote software citation** and metadata workflows
4. **Build bridges** between communities, especially with **Academia**.

Software Heritage

Paste > P169

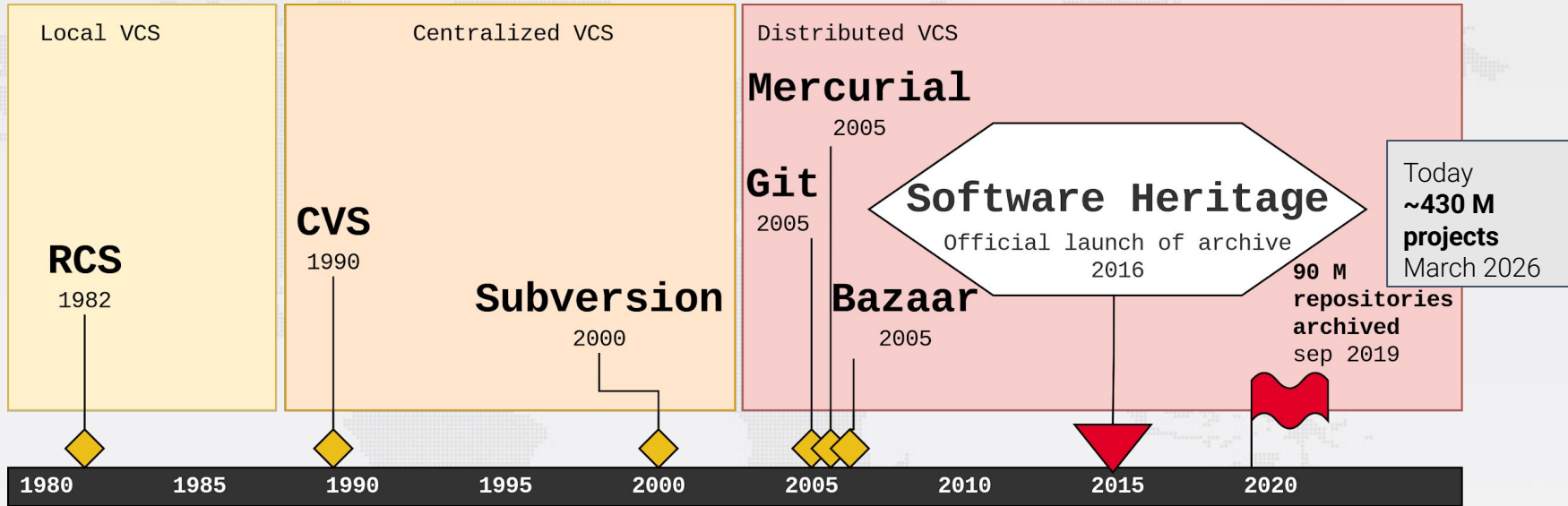
One year plan <https://forge.softwareheritage.org/P169>

Active Public

Authored by moranegg on Jul 18 2017 6:10 PM.

```
1 Goal: Build semantic web of FOSS and promote SWH in citation and metadata workflows
2
3 1. implementation metadata infrastructure/workflow (all tasks are under #Metadata workflow)
4   - [x] strategy and design of metadata component [#T715]
5
6   - storage layer (tables and api entry points)
7     - [x] content level [#T733]
8     - [x] revision level [#T741]
9     - [x] origin level [#T737]
10
11   - indexer layer:
12     - [x] content indexer [#T715]
13     - [x] revision indexer [#T738]
14     - [x] origin indexer [#T1231]
15
16   - fetch extrinsic metadata
17     - [x] with deposit [#T832] + loader-core [#T852]
18     - [ ] when listing [#T833]
19     - [ ] when loading [TBD]
20     - [ ] from external catalogs (e.g libraries.io) [TBD]
21     - [ ] deploy
22     - [ ] document metadata workflow for coders and 'normal' users [TBD]
23
24 2. implementation of metadata tools for detection/extraction/translation
25   - [x] swh-metadata-traslator to CodeMeta schema
26   - [ ] coverage CodeMeta crosswalk table
27   - [x] swh-metadata-detector (heuristic for filenames containing metadata)
28   - [ ] CodeMeta generator
29
30 3. contacts and communication aka metadating
```

Version control in the software forges era



- records changes made to a (set of) source code file (s)
- allows to operate on versions: diff/merge/fork/recover etc.
- essential tool for software development



Free/Libre and Open Source Software

<https://www.gnu.org/philosophy/free-sw.html>

"Open" versus "free"

Free and open-source software (FOSS) or free/libre and open-source software (FLOSS) is openly shared source code that is licensed without any restrictions on usage, modification, or distribution. ^[citation needed] Confusion persists about this definition because the **"free"**, also known as **"libre"**, refers to the **freedom of the product**, not the price, expense, cost, or charge. For example, "being free to speak" is not the same as "free beer". ^[19]

https://en.wikipedia.org/wiki/Open_source



0. RUN



1. STUDY

4^{THE}
FREEDOMS
OF GPL



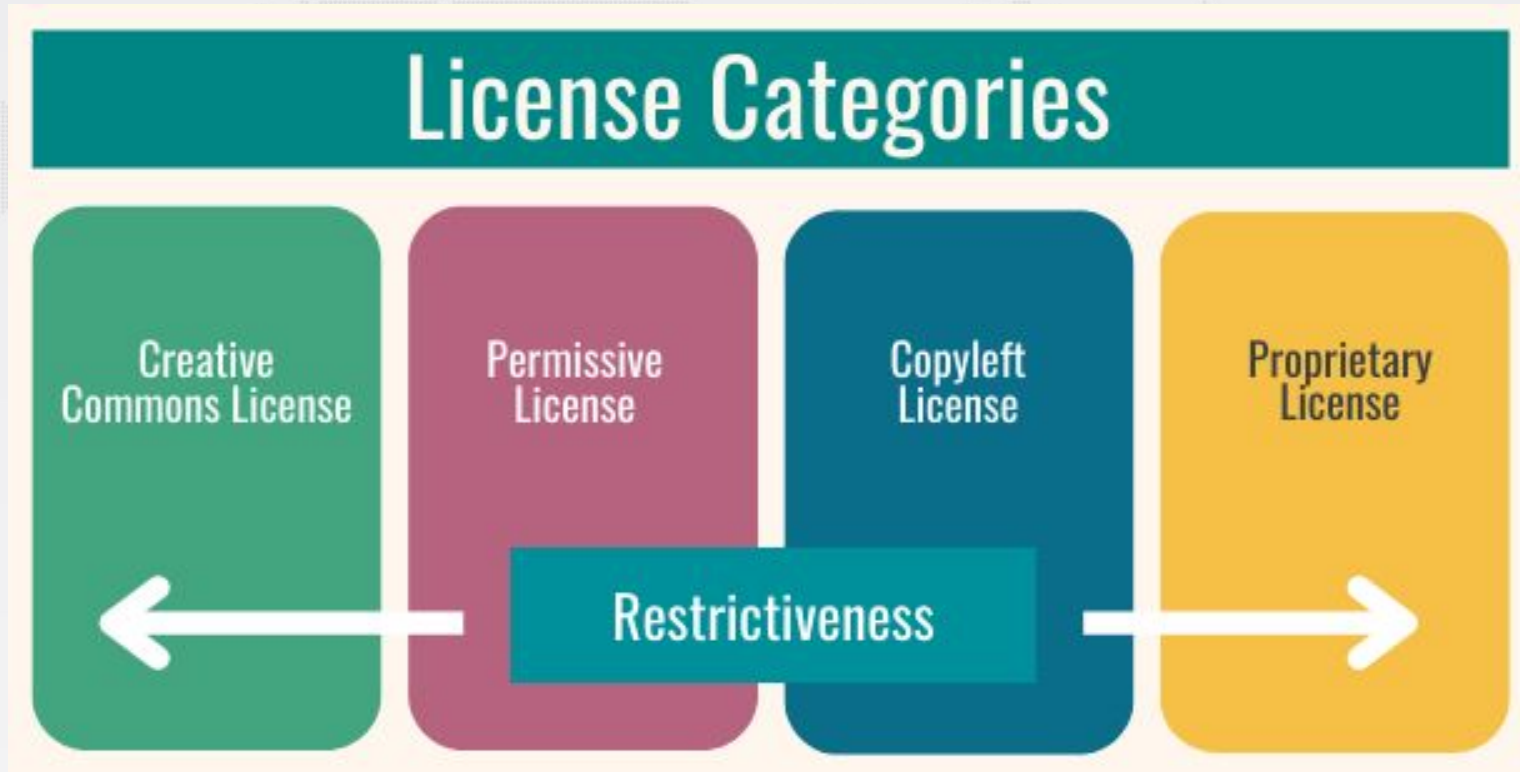
3. SHARE

4. IMPROVE

<https://openclipart.org/detail/246027/four-freedoms-of-free-software>

<https://openclipart.org/detail/320626/craft-beer-glass>

Licensing is a major part in Open Source



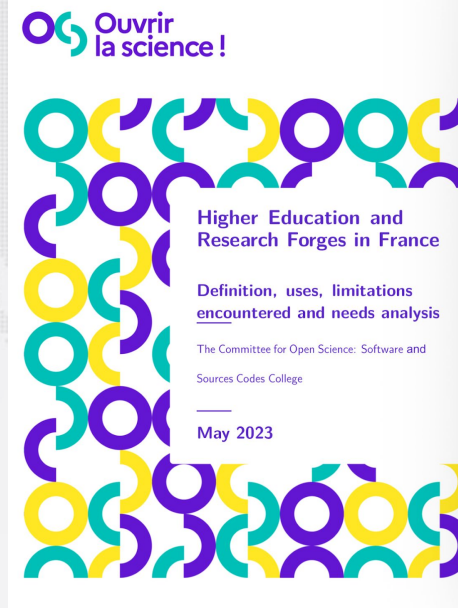
<https://www.tcpipacademy.com/open-source-licensing/>

<https://www.computernetworkingnotes.com/networking-tutorials/software-licenses-types-explained.html>

What are software forges?



*“**Software forges** are mainly designed to manage software-engineering artefacts (such as the source code), the binary files and even the documentation over the entire life cycle. Forges can also be used to collaboratively draft scientific papers or documentation, or even write web pages with the help of tools for processing poorly structured text files ...”*

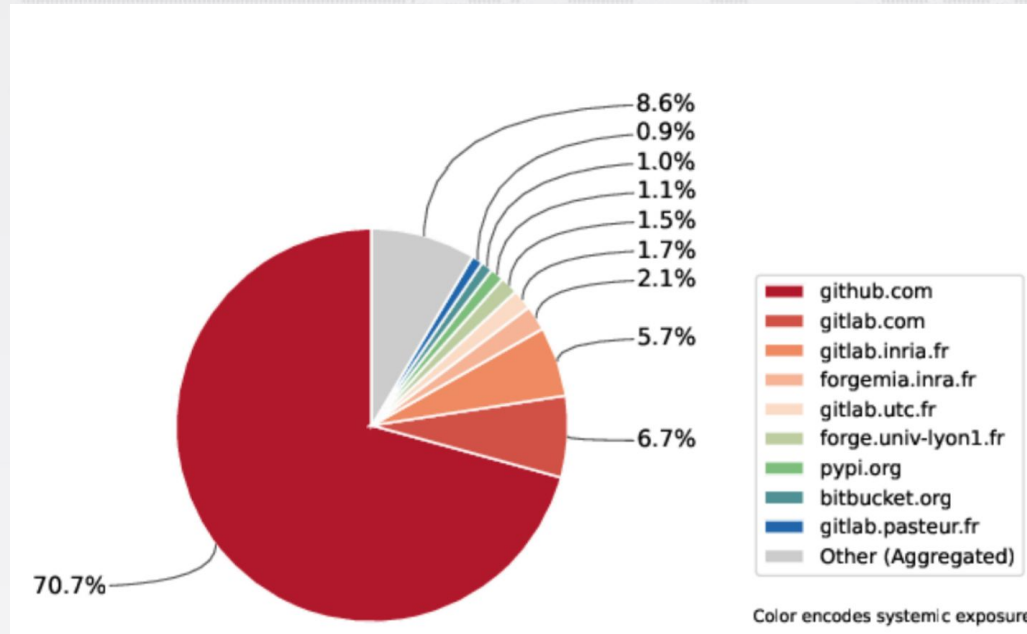


[hal-04208924v1](https://hal.archives-ouvertes.fr/hal-04208924v1)



Tracking researchers commits

Infrastructure concentration (French academia)



Massive concentration in
GitHub > 70%

Di Cosmo, 2025, French Academia Insights report



The commons definitions

The commons

The commons are 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 even when owned privately or publicly.

<https://en.wikipedia.org/wiki/Commons> (retrieved March 1st.)

The software commons

The software commons^[1] consists of all computer software which is available to everyone to use for any purpose. It includes licensed open source software which can be used, studied, modified, and redistributed with few, if any, obligations.^[2] It also includes software in the public domain that is not technically open source software.

https://en.wikipedia.org/wiki/Software_Commons (retrieved March 1st.)

We, the signatories, declare our shared ambition to strengthen Europe's digital sovereignty in an open manner* as a cornerstone of our economic resilience, social prosperity, competitiveness and security.

-

- [illegible]

This Declaration reflects our shared political will to reduce strategic dependencies, strengthen Europe's technological capabilities, preserve democratic resilience, and position Europe as an open, reliable, innovative, and values-based partner in the global digital ecosystem.



Open Source meets Academia | M. Gruenpeter | 13/03/2026 | CC-BY | #12/66

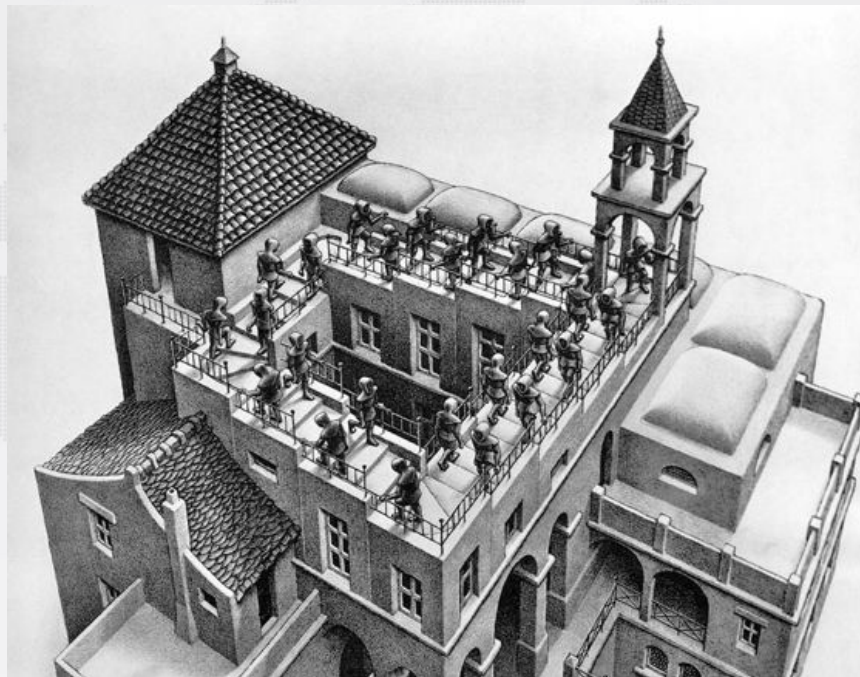
EUtopia or Dystopia for Software



Is **#DigitalSovereignty** even possible?

Is it aligned with the Open Science and Open Source vision?

From **Policies** to **Infrastructures**: the strange loop is infinite.



Ascending and Descending by M. C. Escher

Qu'est-ce qui vous amène aujourd'hui ?



Sur menti: 6268 6204



Le logiciel est-il inclus dans la stratégie **Science Ouverte** de votre institution ?



Posibilités

1. Forte inclusion
2. Faible inclusion
3. Pas de logiciel dans la politique Science Ouverte
4. Pas de politique Science Ouverte

Sur menti: 6268 6204

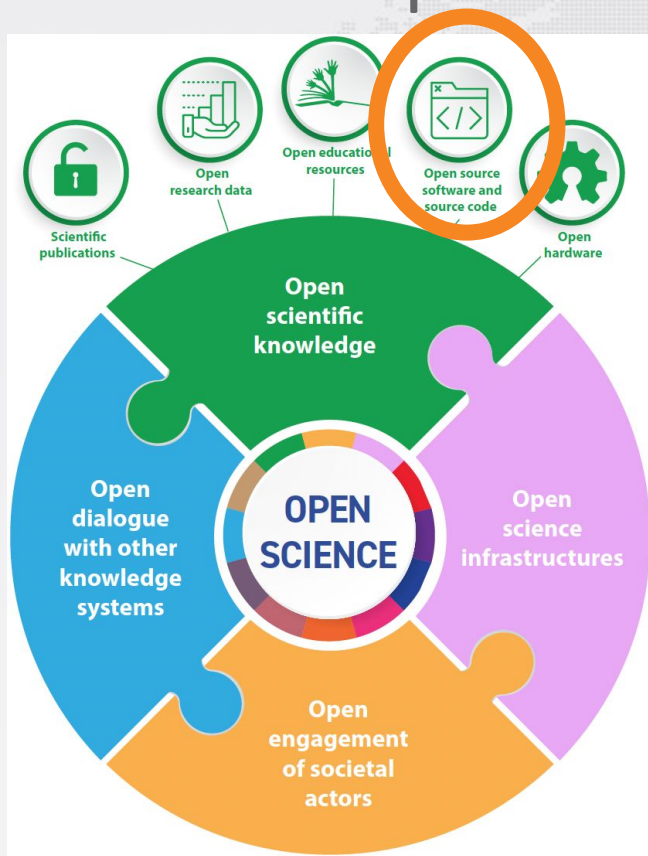


Goals



- ★ **Introduction:** why are we here?
- ★ **Open Science** isn't complete without **Open Source**
- ★ Guided exploration: into the **Software Heritage** archive
- ★ Good practices for **#BetterMetadata**
- ★ The role of libraries: collaborating with **OSP**Os

A vision: Open Science for knowledge freedom



“Building on the essential principles of academic freedom, research integrity and scientific excellence, open science sets a new paradigm that integrates into the scientific enterprise practices for **reproducibility**, **transparency**, **sharing** and **collaboration** resulting from the increased opening of scientific contents, tools and processes.”

[UNESCO Recommendations](#)



Software in Research: A pillar of Open Science

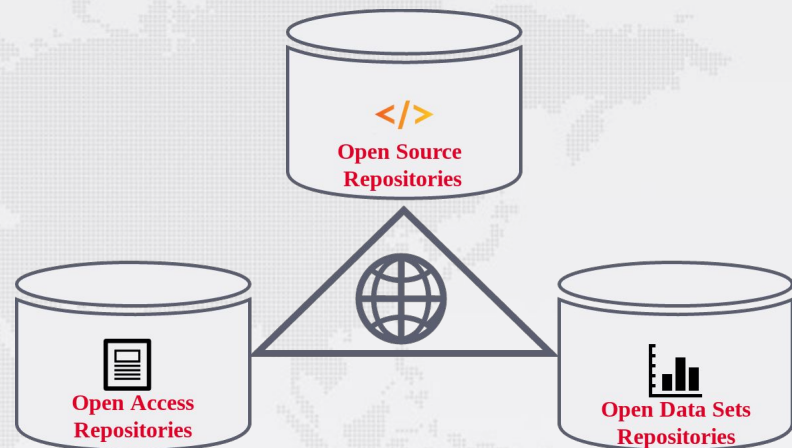


Multiple facets, it can be seen as:

- a tool
- a research **outcome** or result
- the object of research



Tim Berners-Lee the **World Wide Web** was born in Academia, 1989, at CERN



*Three pillars of Open Science
Gruenpeter, Software Heritage CC-BY 4.0 2019*

The Knowledge is in the Source Code



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

Len Shustek, 2006
Computer History Museum

Apollo 11 Guidance Computer (~60.000 lines), 1969

```
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      G0T0P00H      # 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 ...
              CADR      BURNBABY
```

Margaret Hamilton



[Apollo 11 source code in SWH](#) | [Article](#)

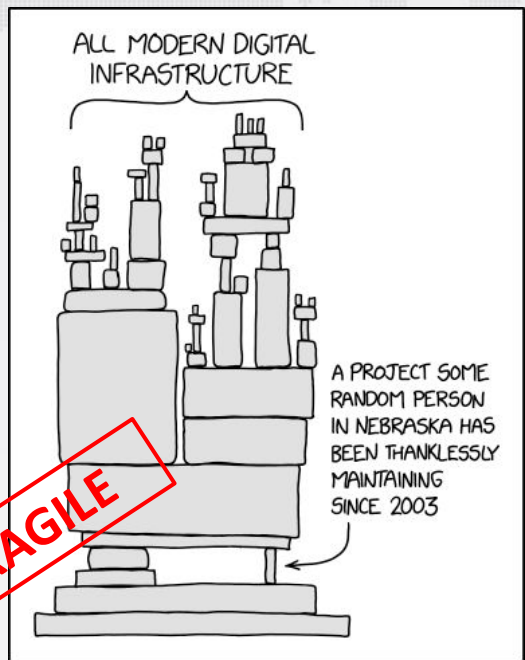


Preserved in the **MIT research library**
digitized by Rob Burkey since 2003

The software stack: More than Research Software

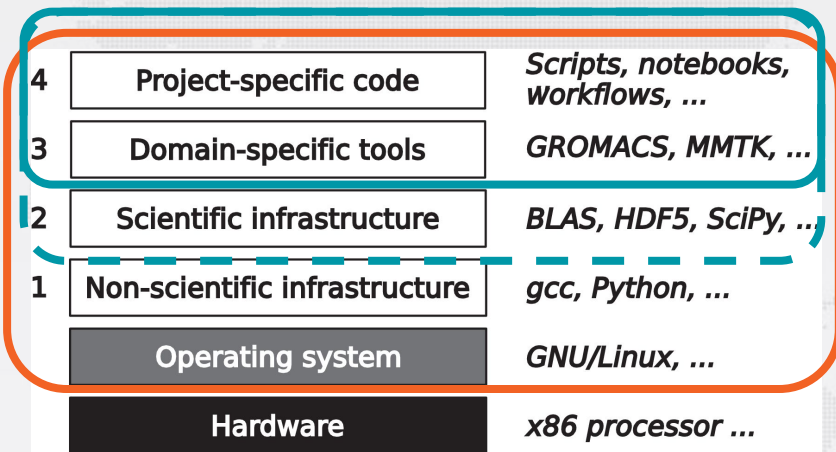


Research Software is a thin layer on top of the global software stack



<https://xkcd.com/2347/>

What's the the Software Collapse?

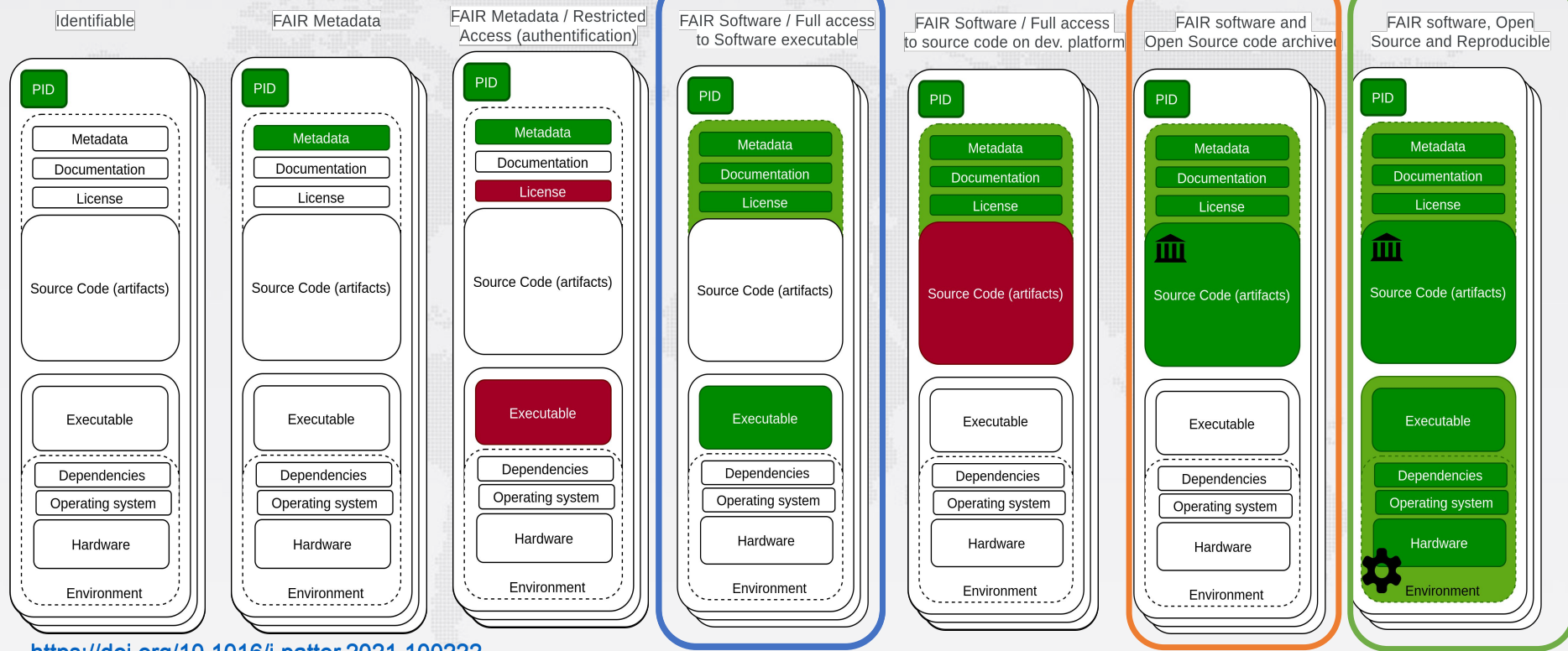


Konrad Hinsén. Dealing With Software Collapse. Computing in Science and Engineering, 2019, 21 (3), pp.104-108. [\(10.1109/MCSE.2019.2900945\)](#). [\(hal-02117588\)](#)

Can be preserved in a:

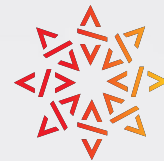
- [Scholarly Infrastructure](#)
- [Universal Source Code Archive](#)

FAIR, Open Source and Open Science?



<https://doi.org/10.1016/j.patter.2021.100222>

Why Open Source is essential to Open Science



Reproducibility: Without open code, results cannot be fully verified or replicated.

Transparency: Algorithms and workflows must be inspectable.

Reusability: Open licenses allow others to reuse, adapt, and extend research tools.

Collaboration: Open development enables global, distributed contribution. Building on communities and governance.

Digital Sovereignty: Institutions choose how to retain control over their research assets

Sources

The Good Governance Handbook <https://ospo-alliance.org/ggi/>

A tale of two 'opens': intersections between Free and Open Source Software and Open Scholarship. SocArXiv. doi:[10.31235/ospf.io/2kxq8](https://doi.org/10.31235/ospf.io/2kxq8)

Defining Research Software: Still a controversy?



Research Software includes source code files, algorithms, scripts, computational workflows and executables that were **created during the research process or for a research purpose.** Software components (e.g., operating systems, libraries, dependencies, packages, scripts, etc.) that are used for research but were not created during or with a clear research intent should be considered software in research and not Research Software. This differentiation may vary between disciplines. The minimal requirement for achieving computational reproducibility is that all the computational components (Research Software, software used in research, documentation and hardware) used during the research are identified, described, and made accessible to the extent that is possible.

Research Software $\neq$ Software in research

Gruenpeter, M., Katz, D. S., Lamprecht, A.-L., Honeyman, T., Garijo, D., Struck, A., Niehues, A., Martinez, P. A., Castro, L. J., Rabemanantsoa, T., Chue Hong, N. P., Martinez-Ortiz, C., Sesink, L., Liffers, M., Fouilloux, A. C., Erdmann, C., Peroni, S., Martinez Lavanchy, P., Todorov, I., & Sinha, M. (2021). Defining Research Software: a controversial discussion (Version 1). Zenodo. <https://doi.org/10.5281/zenodo.5504016>



FAIR4RS

Goals



- ★ **Introduction:** why are we here?
- ★ **Open Science** isn't complete without **Open Source**
- ★ **Guided exploration:** into the **Software Heritage** archive
- ★ Good practices for **#BetterMetadata**
- ★ The role of libraries: collaborating with **OSP**Os

Source code is fragile!



Google Kills Off Google Code

Natasha Lomas @riptari / 10:58 AM GMT+1 • March 13, 2015

1.4 million projects

damage
disaster
malicious
obsolete
dependencies
attack
aging
media
tear
dangling
wear
corruption
encryption
format
deletion
reference
storage

SDTimes

Google begins shutdown of its code repository

Latest News Published: March 12th, 2015 - Michael Pehel

After nine years, Google's open-source code repository, Google Code, started closing shop today by disabling new projects and announcing the permanent shut down of the service by January 26, 2016.

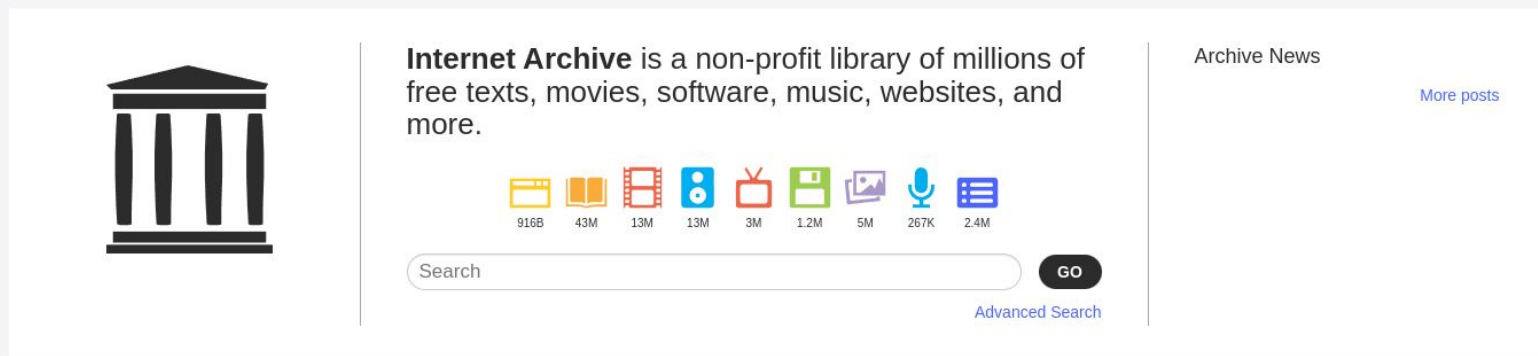
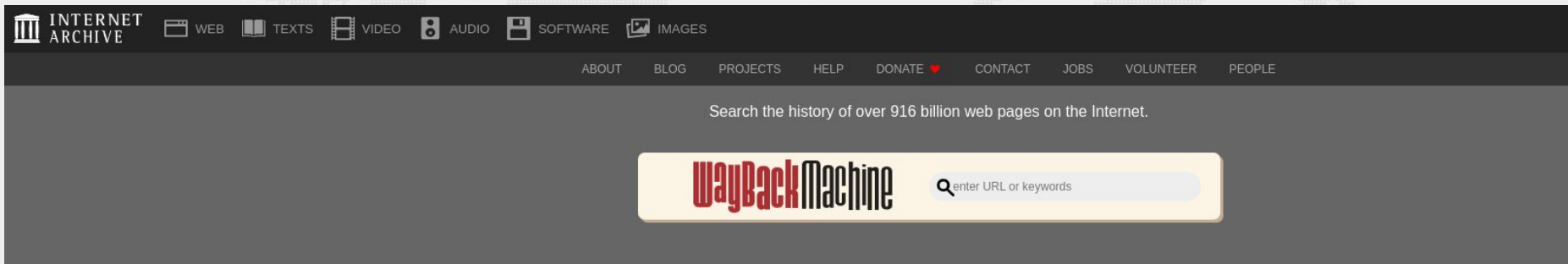
Google Code started as Google's answer to SourceForge, the predominant code repository back in 2006. The reliability of SourceForge was brought into question that year when SourceForge.net's database was hacked and user data was compromised. Problems continued the following summer in 2007 with a temporary service outage in August. The appeal of a repository backed by Google only increased as a result.

delete Google code;
Google Code Shuts Doors



ERROR 404

Archiving the web: the wayback machine



The development history is key



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



We need the complete snapshot

A universal source code archive



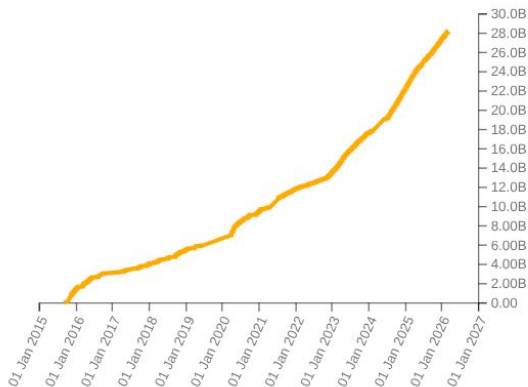
Collect, preserve and share all software source code



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

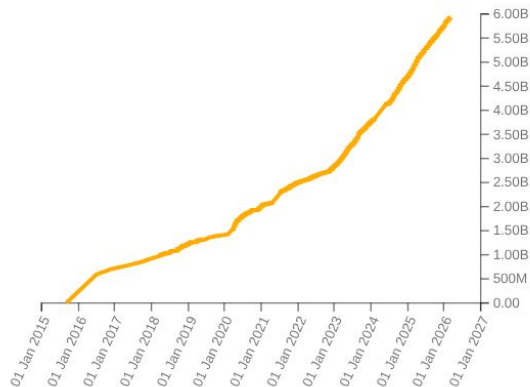
Source files

28,148,748,800



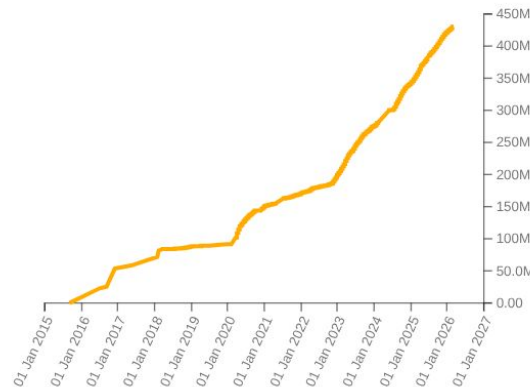
Commits

5,920,096,480



Projects

429,963,770



<https://archive.softwareheritage.org/>

Snapshot: March 2nd 2026

The library of Alexandria of source code



Launched in 2016 by

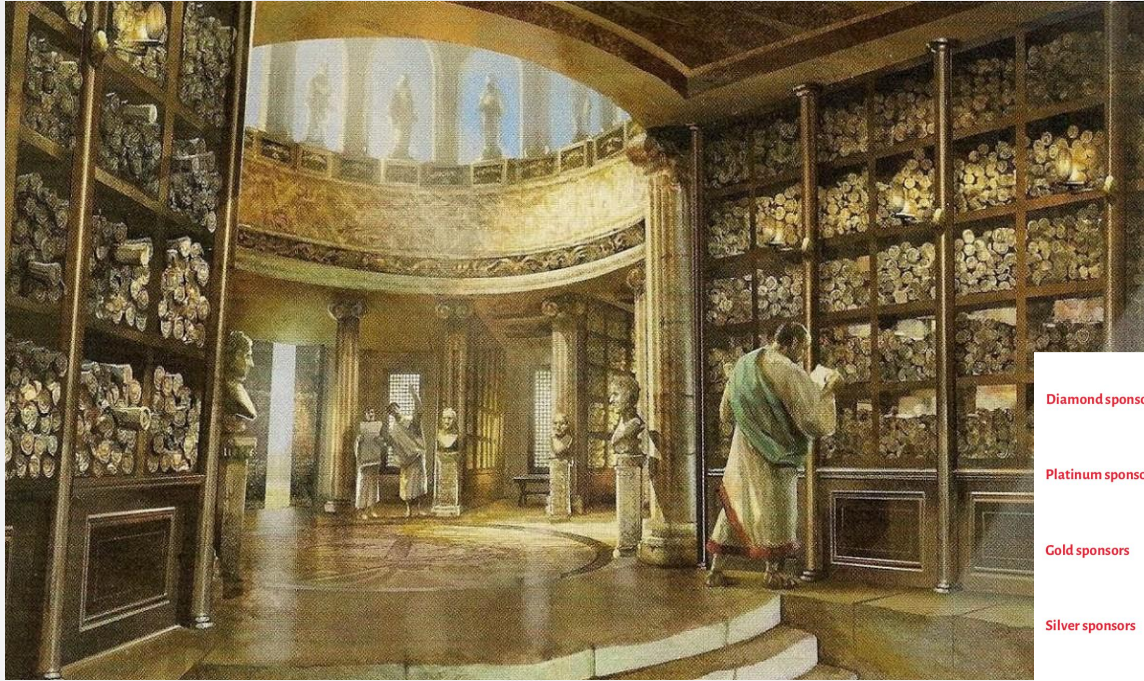
Inria

Hosted SWH Foundation by

Inria

FONDATION

The 2025-2026 Sponsors



In Collaboration with



Diamond sponsors



Platinum sponsors



Gold sponsors



Silver sponsors



Bronze sponsors



Search for your code repository



Full width

Home

Development

Documentation

Donate

Operational

login



Welcome to the Software Heritage archive

Features

-  Search
-  Downloads
-  Save code now
-  Add forge now
-  Help

Search

Enter a SWHID to resolve or keyword(s) to search for in origin urls

- ☒ only show origins visited at least once
- ☒ filter out origins with no archived content
- ☐ search in metadata (instead of URL)

visit type any

... or check our  [Web API](#)

Overview

The long term goal of the Software Heritage initiative is to **collect** all publicly available software in source code form together with its development history, replicate it massively to ensure its **preservation**, and **share** it with everyone who needs it. The Software Heritage archive is growing over time as we crawl new source code from software projects and development forges.

Content

A significant amount of source code has already been ingested in the Software Heritage archive, see the [Archive Changelog](#) for notable changes to the archive over time. It currently includes the following software origins.

Archive without leaving the **forge**



The Browser extension

This tab shows the archival status of the repository

- Green** up to date
- Yellow** not up to date
- Grey** not archived yet
- Red** not archivable (private)

<https://www.softwareheritage.org/browser-extensions/>

Archive a complete **forge** instance



Full width

Home

Development

Documentation

Donate

Operational

Logged in as moranegg, logout

Software
Heritage
Archive



Request the addition of a forge into the archive

Enter a SWHID to resolve or keyword(s) to search for in origin URLs



"Add forge now" provides a service for Software Heritage users to save a complete forge in the Software Heritage archive by requesting the addition of the forge URL into the list of regularly visited forges.

If you submit an Add Forge Now request, please avoid submitting the forge repositories at the same time in [Save Code Now](#), which is designed for the punctual archival of single repositories. We process regularly the Add Forge Now requests, ensuring they will be processed rapidly. If there is a strong deadline due to a risk of source code loss (forge to be closed soon for example), please mention it explicitly in your submission message, so that we will prioritize your request.

Submit a Request

Browse Requests

Help

Forge type*

Supported forge types in software archive.

Forge URL*

Base URL of the forge.

Forge contact name*

Name of the forge administrator.

Forge contact email*

Email of the forge administrator. The given email address will not be used for any purpose outside the "add forge now" process.

☐ I consent to add my username in the communication with the forge.

Comment

Optionally, leave a comment to the moderator regarding your request.



Add forge request

Submit Add Request

Must be
logged in

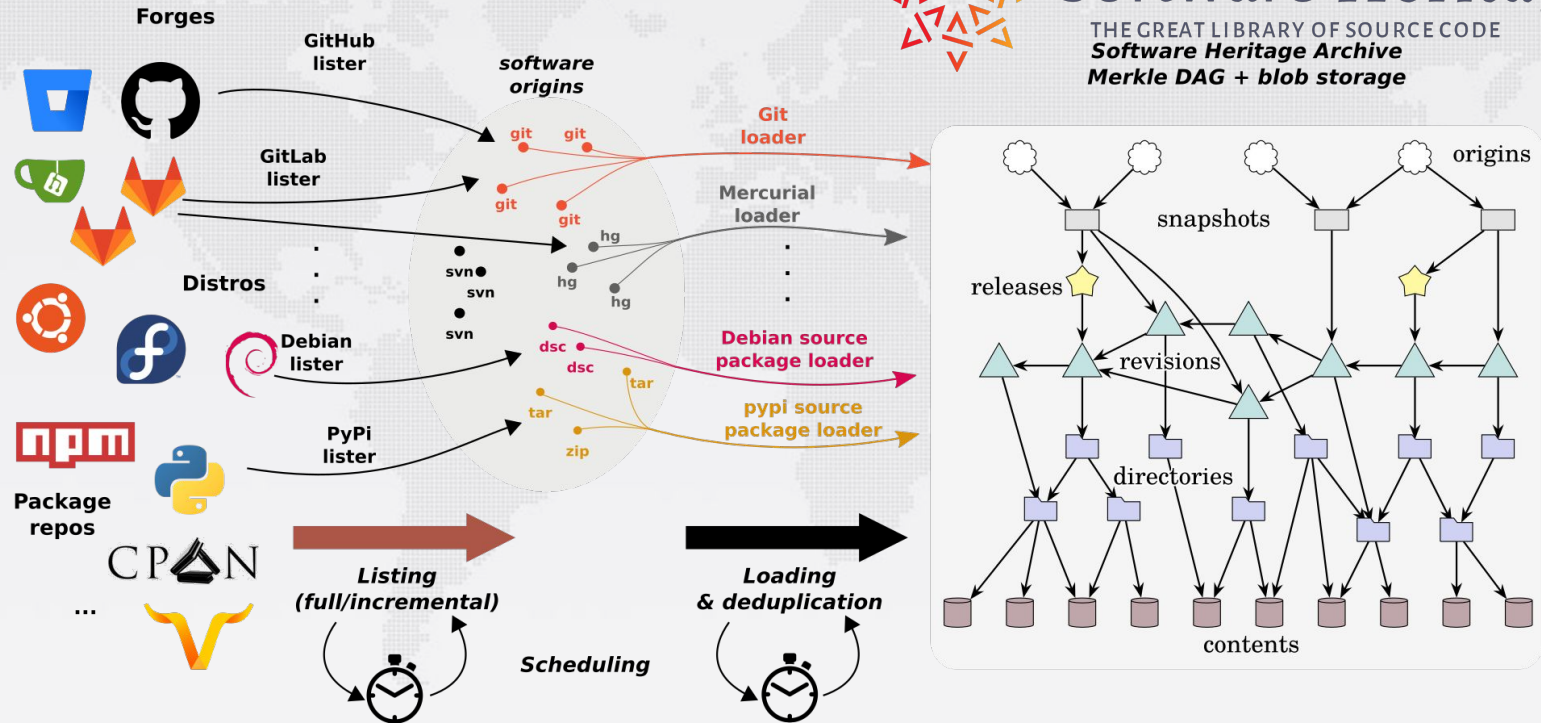
<https://www.softwareheritage.org/browser-extensions/>

A resilient solution: Preserving code from forges



Software Heritage

THE GREAT LIBRARY OF SOURCE CODE
Software Heritage Archive
Merkle DAG + blob storage





Crawling: listing and loading

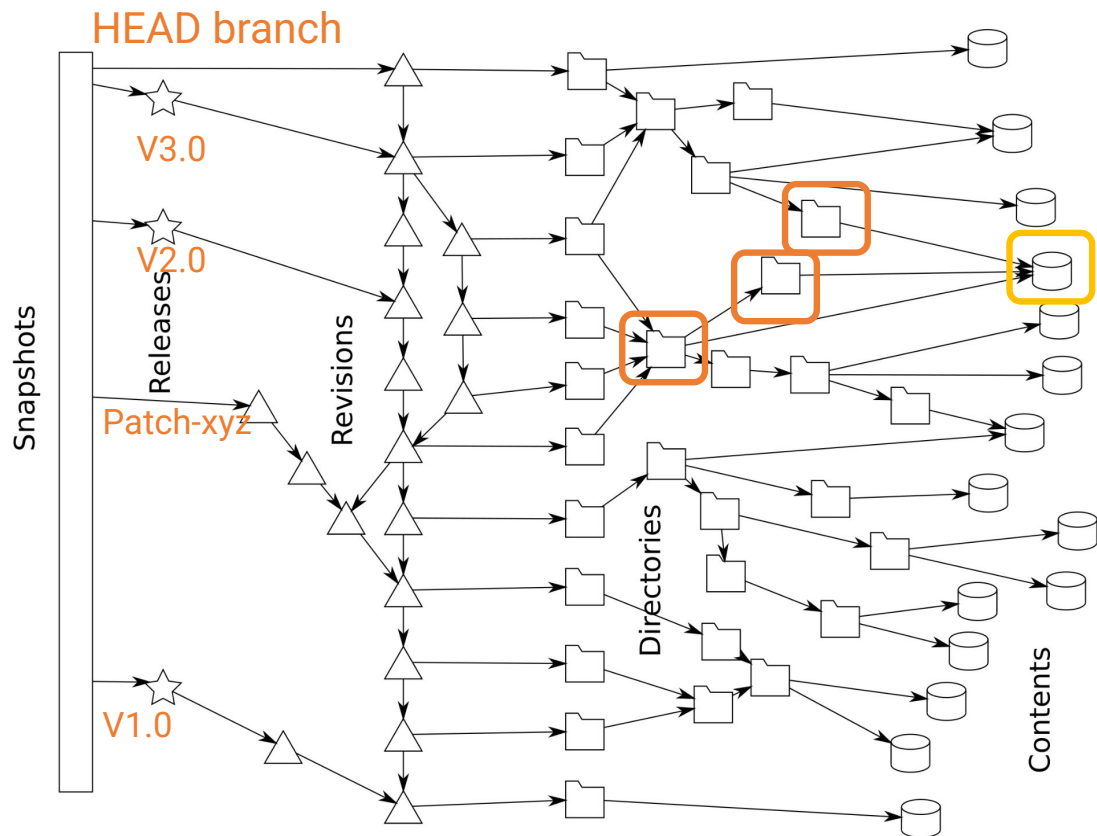
The SWH archive harvests actively source code from different sources and converts all the source code into a single and universal data structure which is an enormous Merkle DAG [[Merkle, 1987](#)].

Crawling is separated into three phases:

1. **listing** software sources,
2. **scheduling** updates and
3. **loading** the software artifacts into the archive.

 6,596 origins	 5,812,433 origins	 56,984 origins
 33,554 origins	 50,523 origins	 31,202 origins
 99,741 origins	 2,135 origins	 146,855 origins
 64,594 origins	 38,070 origins	GitHub 308,309,489 origins
 25,289 origins	 5,917,137 origins	 3,978 origins
 494 origins	 2,373,896 origins	 79,018 origins
 354 origins	 18,932 origins	 1,423 origins
 667,956 origins	 650,849 origins	 73,364 origins
 4,898,191 origins	 5,252 origins	 384,498 origins
 99,390 origins	 199 origins	 76,754 origins
 7,625 origins	 749,476 origins	 175,285 origins
 382,404 origins	 344 origins	

Full development history permanently archived in a **uniform data model**



No
deduplication

Repo github.com/namespace/repo-abc Visited on DD/MM/YYYY



Granularity level does matter

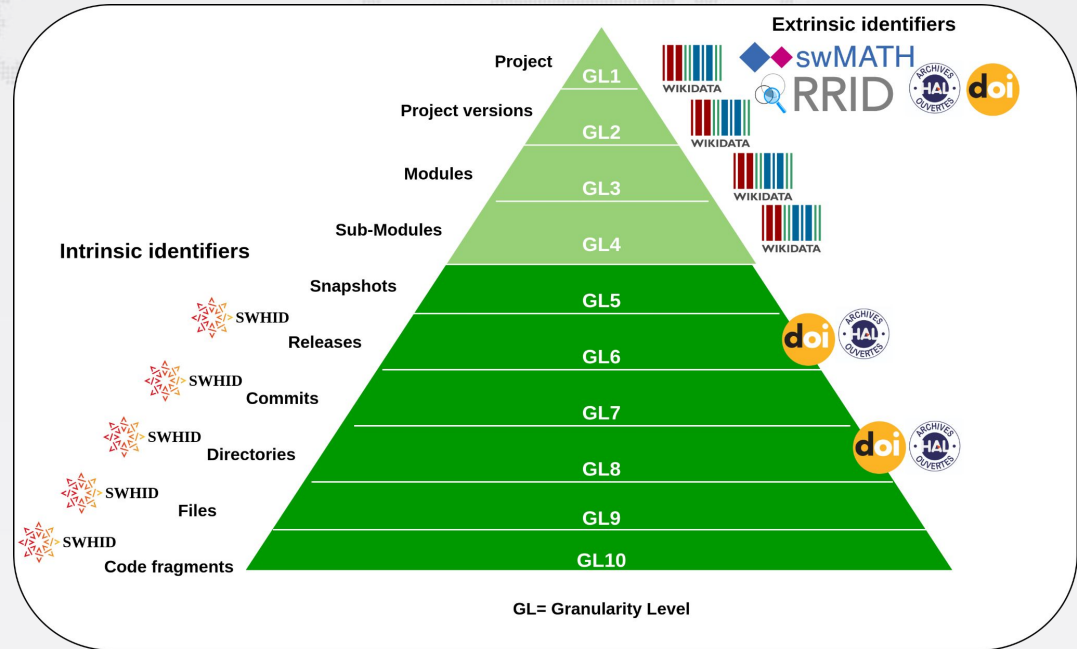
Software concept / project / collection

Description in registry, a homepage or any other form of metadata record

- Project versions (e.g Python2 and Python3)
- Modules
- Sub-modules

Software artifact

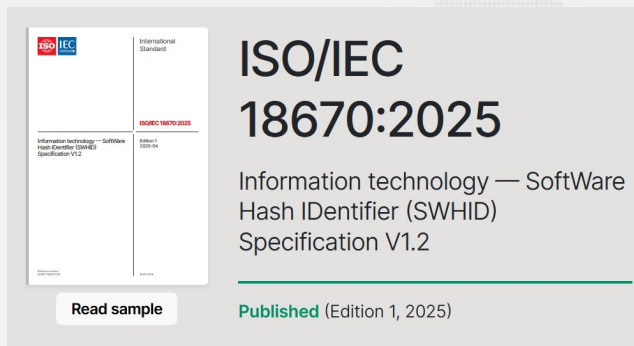
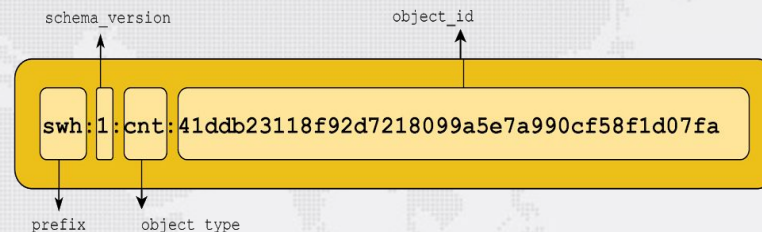
- Executable (download link of binary)
- Software source code
 - ◆ Dynamic artifact - living code repo
 - ◆ Archived copy
 - Snapshot (all branches, all dev history)
 - Release / Package
 - Commit- a specific point in development history
 - Directory
 - File
 - Algorithm





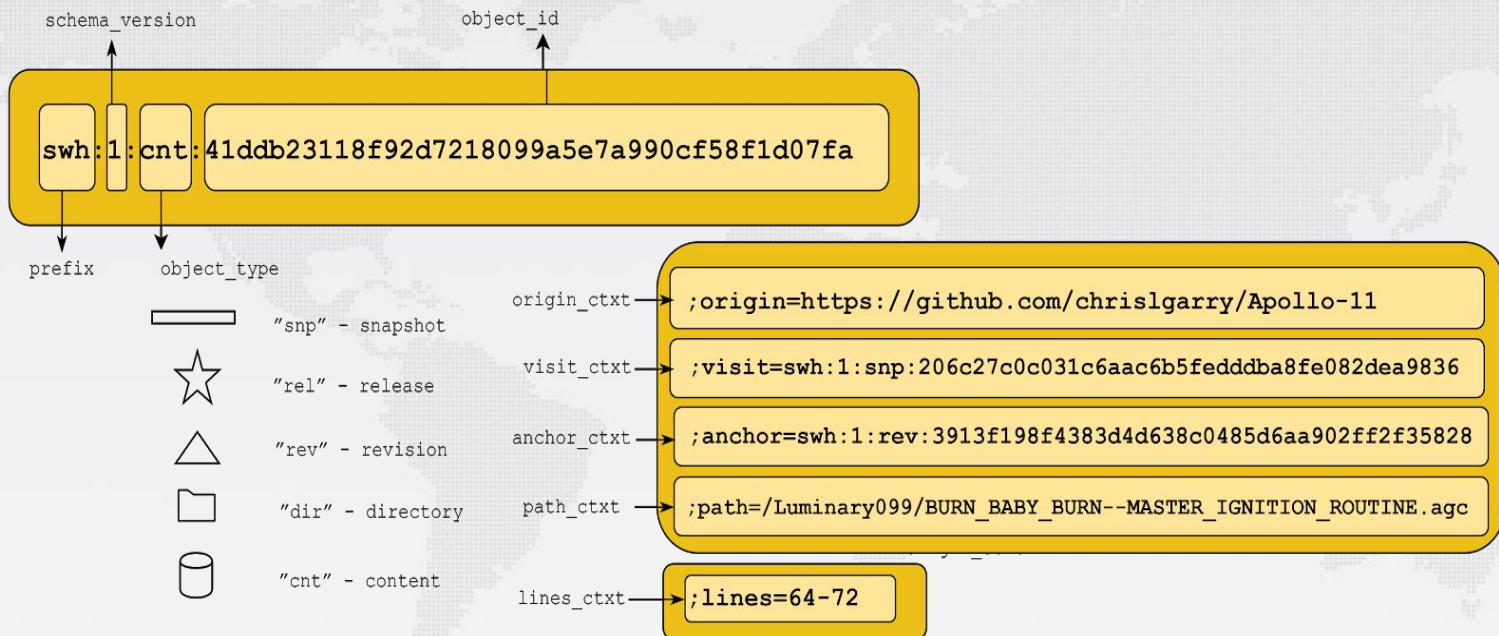
Introducing the SoftWare Hash Identifier

- **Intrinsic**: compute a unique digital fingerprint
- **decentralised**: do not need a registry, only agreement on a standard
- **cryptographically strong** identifiers





SoftWare Hash Identifier with context





Getting a SWHID is easy as ABC

≡ Browse the archive

Enter a SWHID to resolve or keyword(s) to search for it

<https://github.com/rdicosmo/parmap>

28 May 2021, 14:09 UTC

<> Code Branches (52) Releases (8) Visits

★ Branch: HEAD aa10393 / History Download Save again

Tip revision: **f140dbc8b05aa3d341c70436a1920a06df9a0ed4** authored by Roberto Di Cosmo on 02 May 2021, 17:11 UTC
Signal exceptions in the worker to the master process in the simplemapper skeleton.

File	Mode	Size
config		
example		
src		
tests		
.gitignore	-rw-r--r--	38 bytes
AUTHORS	-rw-r--r--	722 bytes
CHANGES	-rw-r--r--	1.8 KB
LICENSE	-rw-r--r--	25.8 KB

Any end-user can retrieve a SWHID:

- no need to create an account on SWH
- no need to be the software author



Extrinsic vs. Intrinsic

Main difference is how the relation between **identifier** and designated object is created and maintained.

Persistence is a key desired property.

	Extrinsic	Intrinsic
Relation	Register	Convention - agreed protocol/naming
Pre-internet	Passport number, ISBN, SSN, etc.	Music / Chemistry notation
Internet era	DOI, Handle, ARK, RRID, etc.	Cryptographic hashes, e.g git, bitcoin, SWHID

A royal wedding: DOI x SWHID



- Journal Image Processing On Line (IPOL, <https://www.ipol.im/>)
- On Software Heritage
 - The article DOI is the **origin**:
(<https://doi.org/10.5201/ipol.2021.286>)
 - The software SWHID:
swh:1:dir:2cb75d8c95eb61d047d89428d0ec40a2286c0311

 IPOL Journal · Image Processing On Line

HOME · ABOUT · ARTICLES · PREPRINTS · WORKSHOPS · NEWS · SEARCH

Image Inpainting using Patch Consensus and DCT Priors
Ignacio Ramirez Paulino, Ignacio Hounie

article demo archive

published 2021-01-09
reference · IGNACIO RAMIREZ PAULINO, AND IGNACIO HOUNIE, *Image Inpainting using Patch Consensus and DCT Priors*, Image Processing On Line, 11 (2021), pp. 1–17 · <https://doi.org/10.5201/ipol.2021.286>

BibTeX info

Communicated by Pablo Arias
Demo edited by Pablo Arias

Abstract

We present an implementation of the PACO-DCT inpainting algorithm. This method is based on maximizing the likelihood of image patches in terms of their DCT coefficients, while requiring consensus on the overlapping patches. The resulting problem is solved as an instance of the PACO framework.

Download

- full text manuscript:  PDF low-res. (577.7kB)  PDF (6.6MB) ^[?]
- source code:  ZIP [SWHID info](#) 

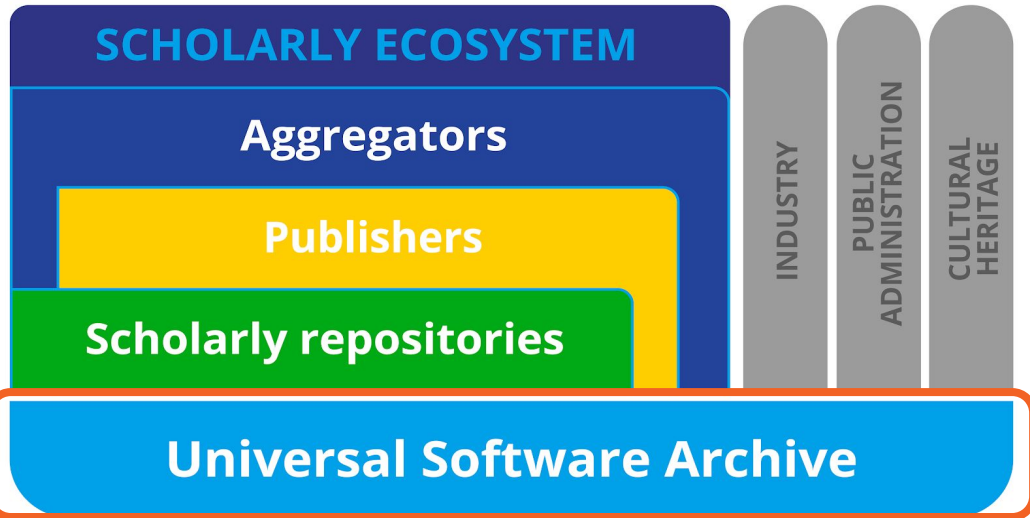
 Software Heritage Archive

```
@softwareversion{sw-ipol.2021.286,
  title   = {{Image Inpainting using Patch Consensus and DCT Priors}},
  author  = {Ignacio Ramirez Paulino, Ignacio Hounie},
  date    = {2021-01-01},
  license = {GPL-3.0-or-later},
  version = {1.0},
  swid    =
  {swh:1:dir:2cb75d8c95eb61d047d89428d0ec40a2286c0311;origin=https://doi.org/10.5201/ipol.2021.286;vis
```

Copy to clipboard

Serving the scholarly ecosystem

Creating connections and mutualizing efforts



Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Software records are deposited and linked on Zenodo

zenodo

Search records...

Communities

My dashboard

Log in

Sign up

Dryad

Published October 22, 2024 | Version v1

Software

Open

Data from: Identification of plankton habitats in the North Sea

Plonus, Rene¹ Floeter, Jens¹

Show affiliations

The definition of an ecological niche makes it possible to anticipate the responses of a species to changing environmental conditions. Broad tolerance limits and a paucity of readily observable niches in the pelagic zone make it difficult to anticipate responses of the plankton community related to anthropogenic or environmental changes. Plankton distributions are closely linked to climate change and shape the seascape for higher trophic levels, so monitoring plankton distributions and defining ecological niches will help to understand and predict ecosystem responses. Here we apply a machine learning autoencoder and a density-based clustering algorithm to high-frequency datasets sampled with a ROTV Triaxus in the North Sea. The results indicate that in this highly dynamic environment, local hydrography prevents niche-based separation of plankton species at the sub-mesoscale, despite the availability of different habitats. Plankton patches were associated with naturally occurring frontal systems and anthropogenically induced upwelling-downwelling dipoles in the vicinity of offshore wind farms (OWFs).

Methods

Physical and biological oceanographic measurements were recorded on different North Sea surveys with the RV Heincke (Knust et al., 2017) using a MacArney TRIAXUS ROTV, complemented by a Video Plankton Recorder (VPR). The TRIAXUS was towed behind the research vessel in an undulating fashion between the sea surface and bottom.

Data was processed using a machine learning Autoencoder and a density-based clustering algorithm HDBSCAN. Analysis and data handling were handled with the statistical software R4.4.0 and Python 3.7.

A detailed description can be found in 'Identification of plankton habitats in the North Sea' (the DOI can be found at 'Related works').

Knust, R., Nixdorf, U. and Hirsekorn, M. 2017 'Research vessel HEINCKE operated by the alfred-wegener-institute', *Journal of large-scale research facilities JLSRF*, 3, pp. A120–A120. doi: 10.17815/jlsrf-3-164.

Files

scripts.zip

scripts.zip

scripts

AE_euclidean.py

README.txt

contour_plots.R

hist_plot.R

9.4 kB

1.2 kB

6.2 kB

449 Bytes

156 VIEWS

31 DOWNLOADS

Show more details

Versions

Version v1

10.5281/zenodo.13375879

Oct 22, 2024

Cite all versions? You can cite all versions by using the DOI 10.5281/zenodo.13375878. This DOI represents all versions, and will always resolve to the latest one. [Read more.](#)

External resources

Archived in

Software Heritage

swh:1:dir:a5cecd7711d5043da4be6af76749ff638f93909

Indexed in

OpenAIRE

Communities

Dryad

Keywords and subjects

habitat maps

Machine learning

North Sea

plankton distributions

plankton-habitat-associations

média
Normandie
CENTRE RÉGIONAL DE FORMATION AUX CARIÈRES
DES MÉTIERS DE LA NORMANDIE

Open Source meets Academia | M. Gruenpeter | 13/03/2026 | CC-BY | #44/66

CC BY



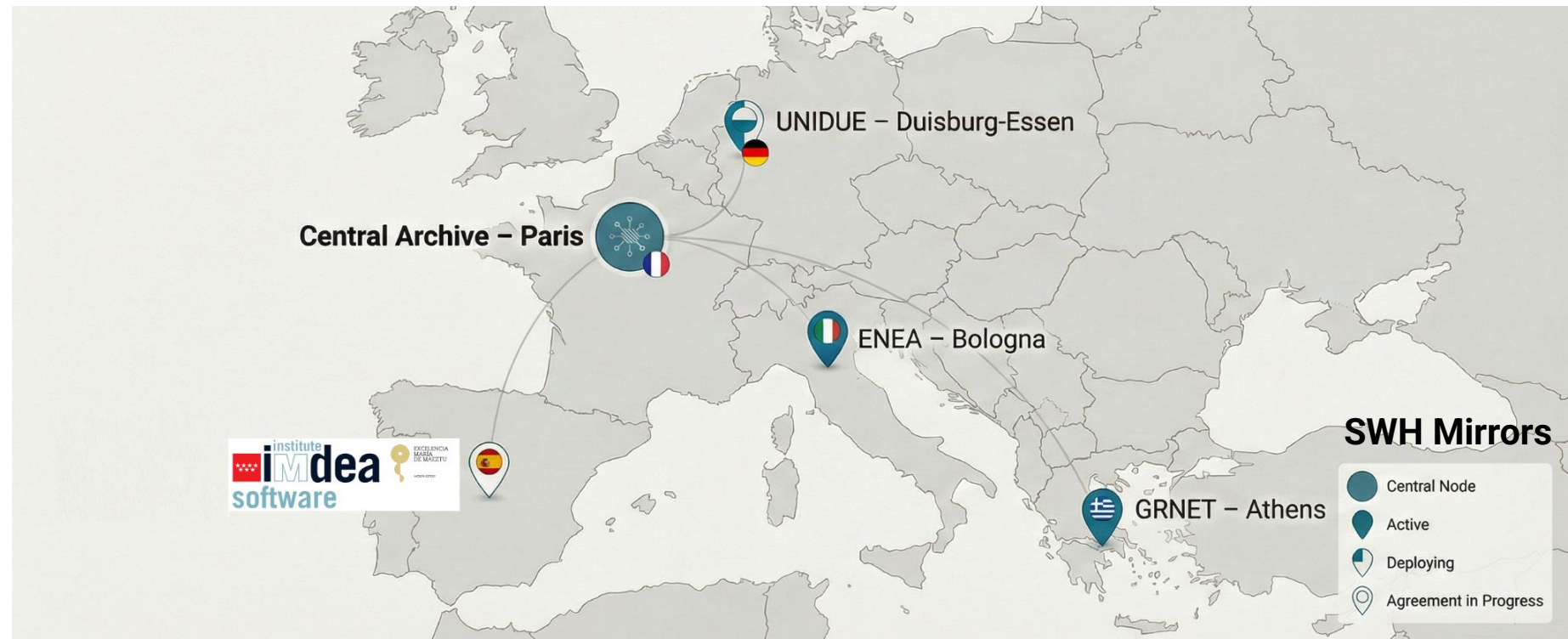
Building for resilience



*"Let us save what remains: not by vaults and locks which fence them from the public eye and use in consigning them to **the waste of time**, but by such a **multiplication of copies**, as shall place them beyond the reach of **accident**."*

— Thomas Jefferson

The Software Heritage Mirrors Network: A distributed multi stakeholders infrastructure



Goals



- ★ **Introduction:** why are we here?
- ★ **Open Science** isn't complete without **Open Source**
- ★ Guided exploration: into the **Software Heritage** archive
- ★ **Good practices for #BetterMetadata**
- ★ The role of libraries: collaborating with **OSP**Os

What is at stake?



#DigitalSovereignty

Archive

- make sure we can access to retrieve the software (reproducibility)

Reference

- make sure we can identify the software artifacts (reproducibility)

Cite (for credit)

- make it rewarding to create software by giving credit to authors (evaluation!)

Describe

- make it easy to discover the software projects (visibility)

Sharing, curating and archiving your source code



Software Heritage

- 1 Prepare your public repository
README, AUTHORS & LICENSE files
- 2 Save your code
<http://save.softwareheritage.org/>
- 3 Reference your work
(full repository, specific version or code fragment)

Step #1: Prepare code with metadata



In the *software source code* itself

- README
- LICENSE
- AUTHORS
- codemeta.json
- package management
 - pom.xml
 - package.json
 - ...
- CITATION.cff
- .About
- ...

The screenshot shows the GitHub web interface for the repository `https://github.com/rdicosmo/parmap`. The file `codemeta.json` is selected, showing its content. The file is a JSON object with the following structure:

```
1 {
2   "@context": "https://doi.org/10.5063/schema/codemeta-2.0",
3   "@type": "SoftwareSourceCode",
4   "license": "https://spdx.org/licenses/LGPL-2.0-only",
5   "codeRepository": "git+https://github.com/rdicosmo/parmap.git",
6   "datePublished": "2011-07-18",
7   "dateModified": "2022-01-03",
8   "issueTracker": "https://github.com/rdicosmo/parmap/issues",
9   "name": "Parmap",
10  "version": "1.2.5",
11  "applicationCategory": "Parallel computing",
12  "developmentStatus": "active",
13  "referencePublication": "https://doi.org/10.1016/j.j.procs.2012.04.202",
14  "programmingLanguage": [
15    "OCaml"
16  ],
17  "operatingSystem": [
18    "Linux",
19    "MacOS"
20  ],
21  "relatedLink": [
22    "https://opam.ocaml.org/packages/parmap/"
23  ]
24 }
```

Human readable (e.g README)

Machine actionable (e.g codemeta.json)

Step #2: Save ~~your~~ any code to SWH



<https://save.softwareheritage.org/>

Save code now

 Software Heritage Archive

Features

- Search
- Downloads
- Save code now**
- Help

Save code now

Enter a SWHID to resolve or keyword(s) to search for in origin URLs

🔍

You can contribute to extend the content of the Software Heritage archive by submitting an origin save request. To do so, fill the required info in the form below:

Origin type

git **1**

Origin url **2**

Submit **3**

Help

Browse save requests

A "Save code now" request takes the following parameters:

Documentation

<https://www.softwareheritage.org/how-to-archive-reference-code/>

 média
Normandie
CENTRE RÉGIONAL DE FORMATION AUX CARRIÈRES
DES MÉTIERS PUBLICS DE NORMANDIE

Open Source meets Academia | M. Gruenpeter | 13/03/2026 | CC-BY | #51/66

 CC BY

Step #3: Reference and cite code with the SWHID



Full width

HomeDevelopmentDocumentationDonate

Operationallogin

Software Heritage Archive

Browse the archive

Enter a SWHID to resolve or keyword(s) to search for in origin URLs

visit typegit

CodeBranches (15)Releases (62)Visits

Branch: HEADb7d8a88 /

Tip revision: 12879609b7fdc9fc384e039c6954538563175a2a281a85d7d6b3133Merge pull request #19346 from kronbichler/feature/2020-02-02

File

.githubbundledcmakecontribdocexamplesincludemodulesourcestests

PermalinksCitations

This interface enables to generate software citations, provided that the root directory of browsed objects contains a [citation.cff](#) or [codemeta.json](#) file. Select below a type of object currently browsed in order to generate citations for them.

directoryrevisionsnapshot

FormatBibTeXGenerate software citation in BibTeX format (requires biblatex-software package)

```
@softwareversion{swh-dir-b7d8a88,
  author = "Arndt, Daniel and Bangerth, Wolfgang and Davydov, Denis and Heister, Timo and Heltai, Luca and Kronbichler, Martin and Maier, Matthias and Pelteret, Jean-Paul and Turcksin, Bruno and Wells, David",
  doi = "https://doi.org/10.1016/j.camwa.2020.02.022",
  title = "The deal.II finite element library: Design, features, and insights",
  swhid =
    "swh:1:dir:b7d8a881065f0d783748d266dc5e745d277f6962;origin=https://github.com/dealii/dealii;visit=swh:1:snp:33245829926e5538563175a2a281a85d7d6b3133;anchor=swh:1:rev:12879609b7fdc9fc384e039c69545e518d606afa"
}
```

Browse citation metadata

Copy citation

The Research Software MetaData guidelines



FAIR4RS

SIRS report

The RSMD seven Aspects

1. General Metadata Requirements

2. Accessibility & preservation

3. Reference & identification

4. Description & classification

5. Attribution & credit

6. Reuse, licensing & legal aspects

7. Re-execute: Dependencies & execution environment

A = Archive

A = Accessible

R = Reference

F = Findable

D = Describe

I = Interoperable

C = Cite

R = Reusable

<https://github.com/FAIR-IMPACT/RSMD-guidelines>
<https://fair-impact.github.io/RSMD-guidelines/>

eosc | FAIR-IMPACT



Questions sur les partenaires

Sur menti: 6268 6204



Goals



- ★ **Introduction:** why are we here?
- ★ **Open Science** isn't complete without **Open Source**
- ★ Guided exploration: into the **Software Heritage** archive
- ★ Good practices for **#BetterMetadata**
- ★ **The role of libraries: collaborating with OSPOs**



Research libraries have a pivotal role

Software preservation requires a global and coordinated effort.

The long road ahead:

- availability,
- findability,
- traceability,
- reproducibility

Supporting the supporters: The Archives and Libraries Interest Group (ALIG)



Open infrastructure

Contribute to a 100% Open Source Infrastructure, valuing users input and support



Tailored Support

Receive assistance for your software archival questions, plans and projects



Community

Participate in an inclusive, innovation driven community

ALIG Members



Sant'Anna
School of Advanced Studies – Pisa



Canadian Research Knowledge Network
Réseau canadien de documentation pour la recherche



Konsortium der Schweizer Hochschulbibliotheken
Consortium des bibliothèques universitaires suisses
Consorzio delle biblioteche universitarie svizzere
Consortium of Swiss Academic Libraries

couperin.org

Consortium unifié des établissements universitaires et de recherche pour l'accès aux publications numériques



<https://www.softwareheritage.org/support/members-alig/>

Who are the stakeholders?



Researchers

- find and access useful software
- get credit for software
- verify/reproduce results

Laboratories/teams

- track contributions
- produce reports
- maintain web pages



Research Organization

know its software assets for:

- technology transfer,
- impact metrics,
- strategy

Curators

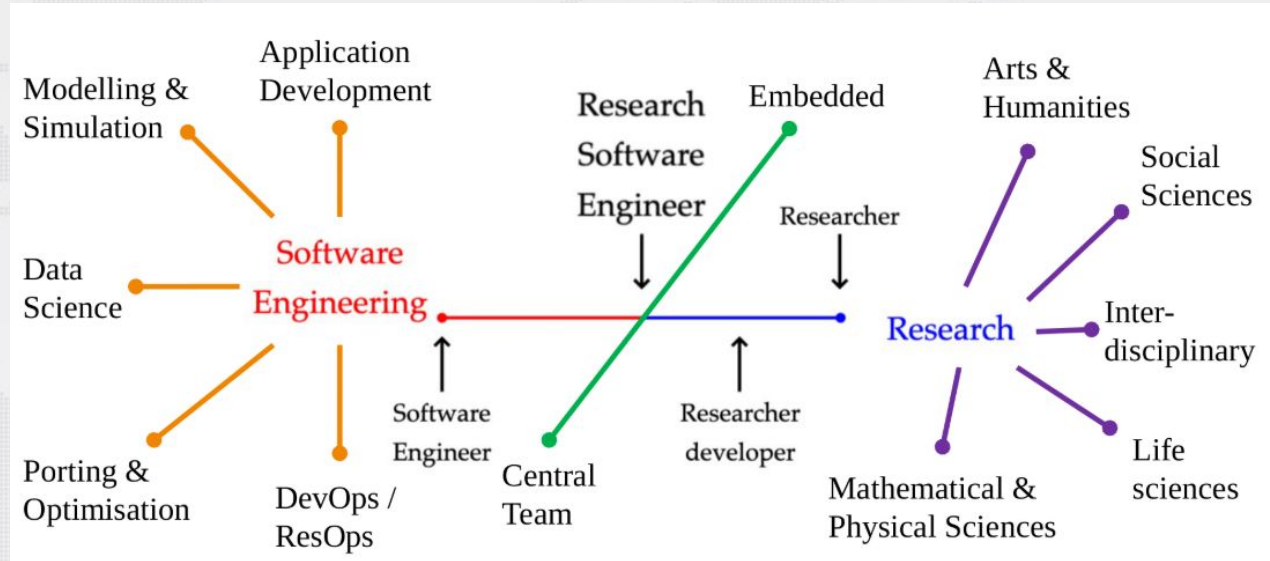
- verify and curate metadata
- provide documentation on software curation
- train researchers for quality metadata

What about Research Software Engineers?



RSEs

- central role for creating and maintaining Research Software
- building bridges between Academia and Software Ecosystems
- support teams with technology



Chue Hong, Neil (2023). Is Research Software Engineering coming of age?. figshare. Presentation.
<https://doi.org/10.6084/m9.figshare.24078054.v5>

OSPPO for Open Source Programme Office

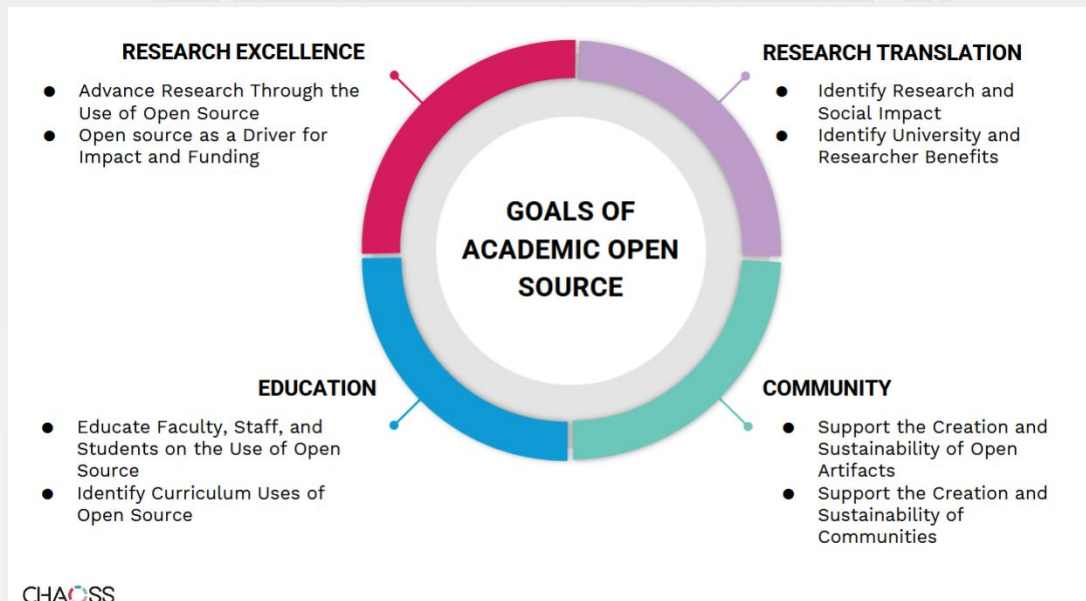


OSPPO

- central role for Open Source strategy
- building bridges between Academia and Open Source
- support teams with technology

<https://code.gouv.fr/fr/blog/definition-ospo/>

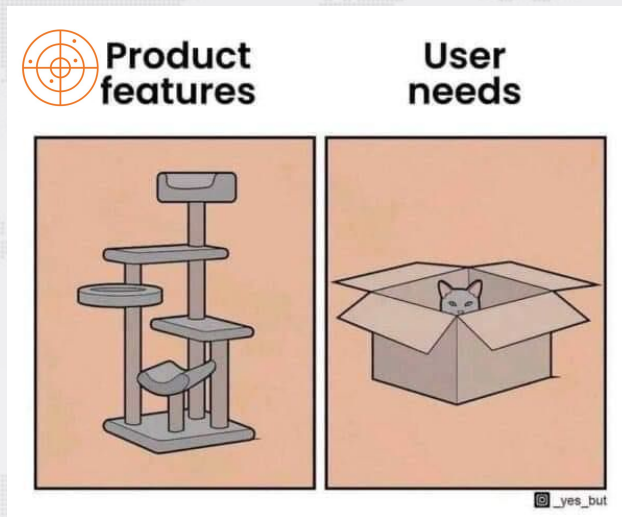
An Open Source Program Office (OSPPO) is an entity within an institution that **defines and implements an open source strategy.**



(C. Dillon, 2025)

[Personas full description](#)

OSPO-RADAR: A layer on top of the SWH archive



“Open-source software development is an essential piece of the science ecosystem that drives and sustains scientific discovery,” said **Joshua M. Greenberg**, Program Director, [Sloan Foundation](#).

“Software Heritage’s OSPO-RADAR project will make visible that influence by giving research institutions a set of tools to better manage, document, and share their open-source work.”



ALFRED P. SLOAN
FOUNDATION

Designing a product with the OSPOs Community

The MVP capabilities

- Create an **institution collection**

- **Batch import;**

- **Signal code repo**

- **Management dashboard:** to view/filter/curate;

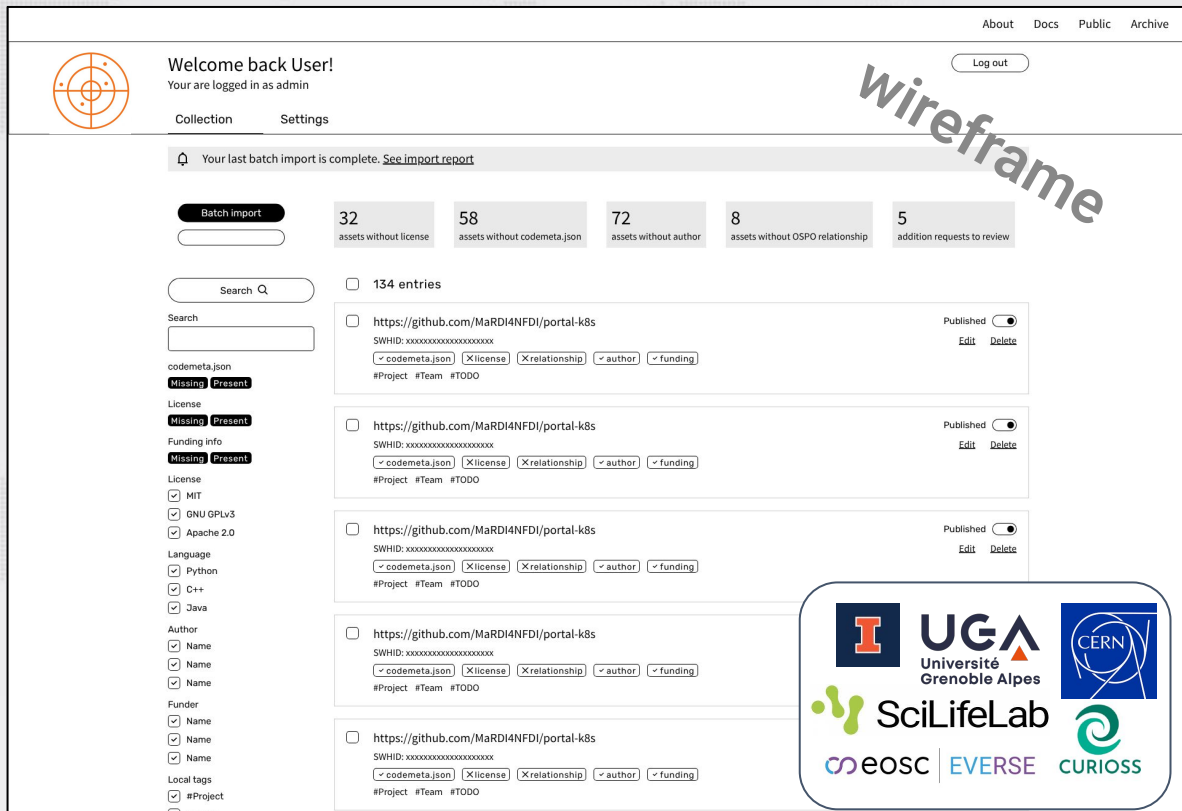
- A public **front-shop** that can be injected in institutional website through an **API**.

Funded by



ALFRED P. SLOAN
FOUNDATION

[Full specifications](#) are public



Software Heritage as a Metadata-broker



Leveraging the CodeMeta vocabulary

- A subset of **schema.org**
- Community governed project

The CodeMeta Project

Software Heritage has the capacity to

- Ingest metadata
- Extract metadata
- Deliver consolidated views to stakeholders

Wireframe

About Docs Public Archive

Welcome back User!
You are logged in as admin

Collection Settings

Metadata extracted from repository

name: example
last modified: 12/13/2019
license: MIT
programming languages: Python
authors:
ORCID 1
ORCID 2
related publications:
DOI

Close code X

Relationship
☒ Created ☒ Contributed ☐ Used ☐ Referenced

Local tags
Project X Topic X

Note

Delete from collection Save

09 December 2025, 12:06:43 UTC

<> Code Branches (215) Releases (0) Visits

★ Branch: HEAD df7196 / History Download

Tip revision: 093ec2a0bd3ae294cdf6034dda9a3af674f03c7 authored by Eloi Ferrer on 09 December 2025, 12:06:23 UTC
deploy wiki base image v1.45.5

File	Mode	Size
github		
apps		
charts		
distrobase		



RSEs and **OSPOs** in Academia
can support **Librarians**

And connect *Academia* to
the rest of the *software world*

Join the mailing-list to follow on news on
new features for OSPOs

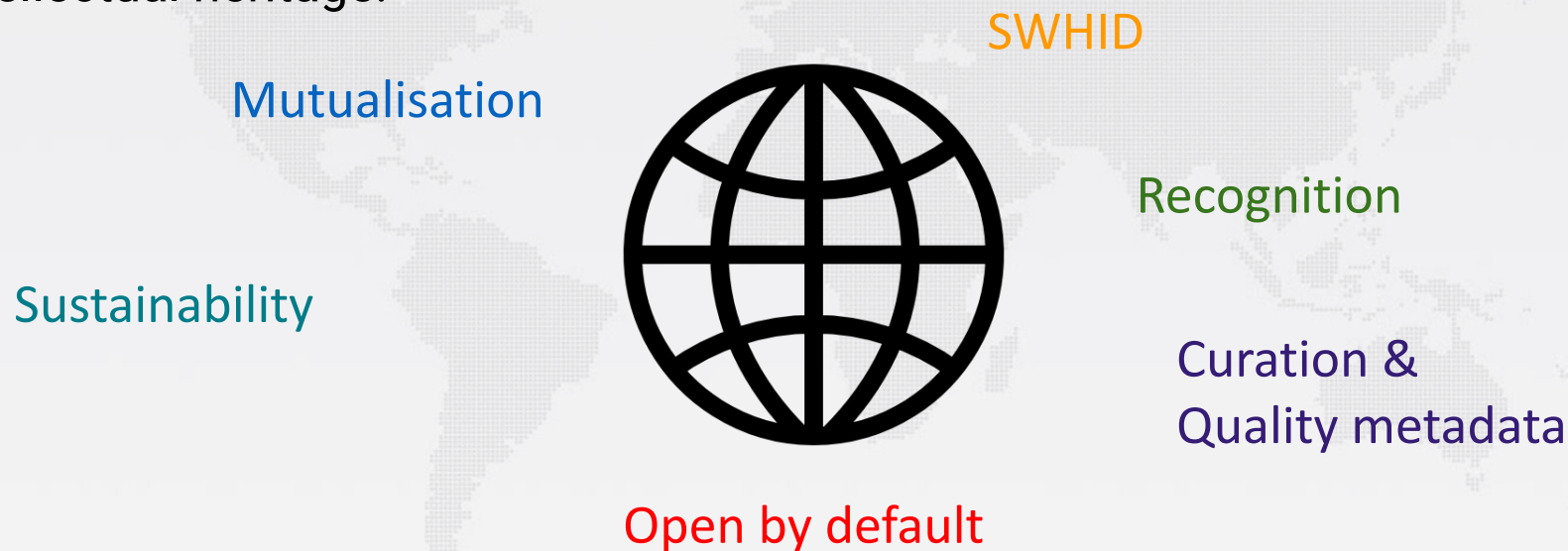
◆ swh-ospo-radar@inria.fr



Building together interoperable Ecosystems



Software source code recognition requires a **supported** and **coordinated** effort. We need to treat code not just as a tool, but as a core pillar of our intellectual heritage.





Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

Thank you

morane@softwareheritage.org

<https://www.softwareheritage.org/>

Browse the **source code archive**

<https://archive.softwareheritage.org/>

Become a **member**

<https://www.softwareheritage.org/support/members-align>

Volunteer as an **ambassador**

<https://www.softwareheritage.org/ambassadors/>

Subscribe to the **newsletter**

<https://www.softwareheritage.org/newsletter/>



The Team

© Inria / Photo B. Fourier



The Community