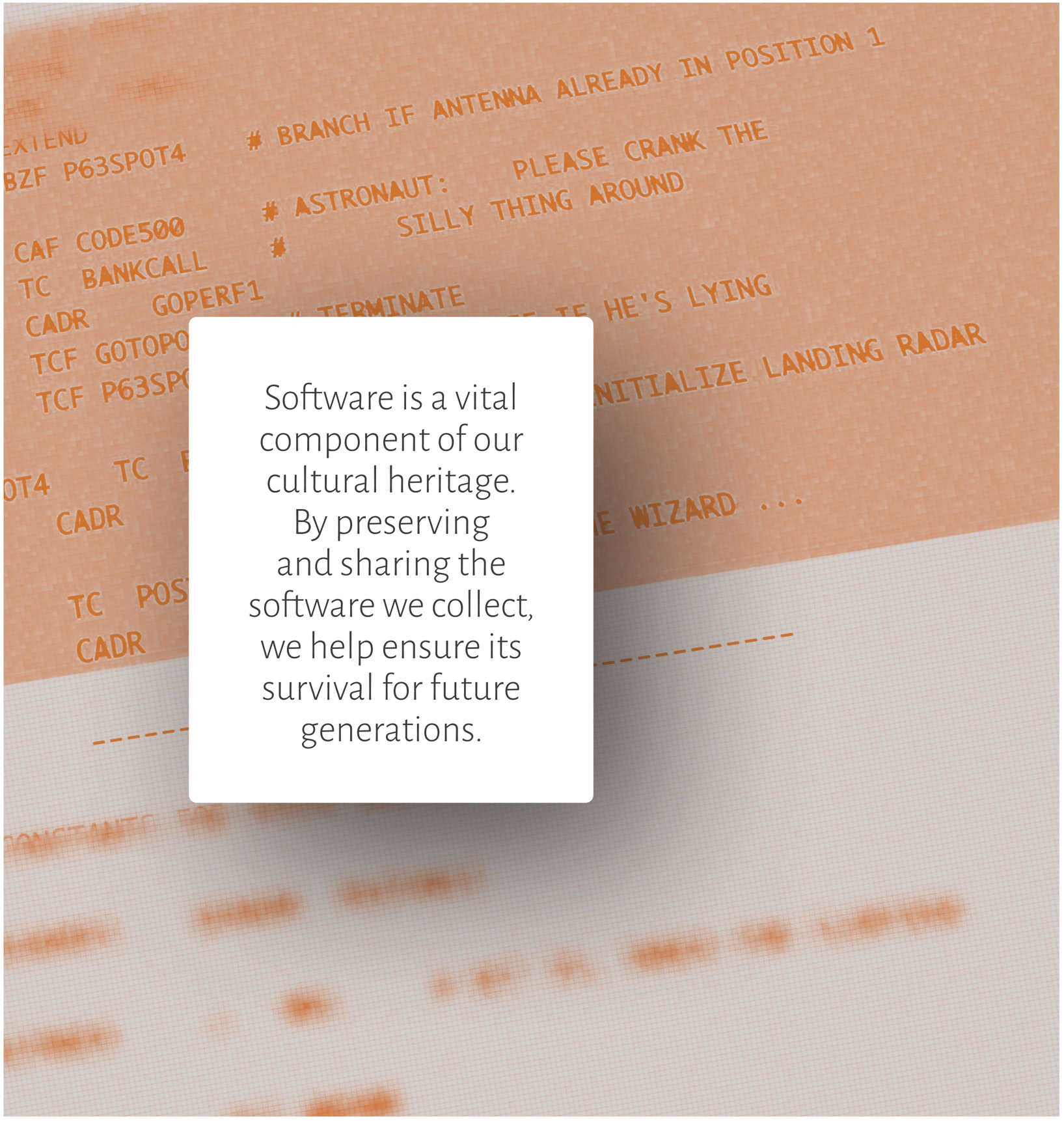
Software Heritage

Annual Report 2024

Collecting, preserving and sharing software source code since 2015



Foreword

As we close the chapter on 2024, we reflect on a year of milestones and progress in Software Heritage's mission to preserve and share the collective knowledge embodied in source code. Software is a vital component of our cultural heritage. By preserving and sharing the software we collect, we help ensure its survival for future generations.

The archive now hosts over 22 billion unique source files from more than 340 million projects, reinforcing its role as a cornerstone of open science, cultural preservation, and technological innovation. Each software artifact is assigned a unique, cryptographically strong identifier: the Software Heritage Identifier (SWHID). Designed for the archive's needs and adopted across academia and industry, the SWHID guarantees content integrity. Years of effort have culminated in a precise specification, now advancing in the ISO/IEC standardization process.

Our commitment to open science deepened through collaborations with repositories like Zenodo and HAL, publishers including Dagstuhl, eLife, Episcience, and aggregators like swMATH and OpenAIRE. Partnerships with initiatives like EOSC and RDA, and projects like FAIR-IMPACT and SoFAIR, championed transparency and reproducibility, underscoring software's pivotal role in research. Through the Archives and Libraries Interest Group (ALIG), we're empowering librarians to support software citation and preservation. Together, we've laid a strong foundation for integrating software in the scholarly ecosystem.

The archive provides a trusted infrastructure to ensure the availability, integrity, and traceability of source code, with applications ranging from industry best practices to AI. It offers the simplest solution for open-source distribution compliance and the detection of leaked code, and it provides a unique platform for improving the transparency and security of the software supply chain. We have engaged directly with the Open Regulatory Compliance Working Group to bring the archive to bear in helping open source to comply with the Cyber Resilience Act. We also started work with a broad group of collaborators to expand the archive and build a code commons that will be a foundation for more transparent and accountable AI models.

Software Heritage's long-term mission is driven by the support of our members, partners, sponsors, ambassadors, and the broader community. As we look to 2025, we invite you to join us. Together, we're preserving source code and building revolutionary infrastructure to serve humankind for generations to come.

Roberto Di Cosmo
Co-founder & CEO
Software Heritage

© CCO | Suginami https://commons.wikimedia.org/wiki/File:Tianjin-BinhaiXinqu_Library.jpg

We collect publicly available source code from numerous software projects and track their ongoing development.
To date, our archive safely preserves:

- **22,437,196,971** Source files
- **4,721,480,727** Commits
- **343,604,069** Projects

Table of contents

6 About

8 Sponsors

10 Mission

12 Software Heritage in a nutshell

15 Tech highlights

16 Software Hash Identifier

18 Services

20 Culture and education

22 The software pillar of Open Science

30 Industry

33 Mirrors

36 Collaboration and community

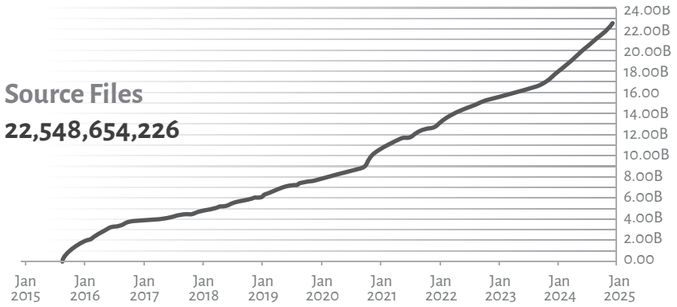
38 Team

39 Ambassadors

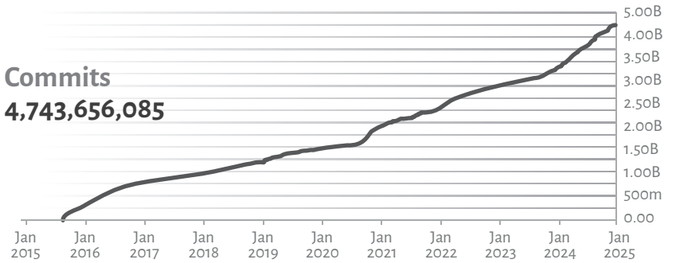
About us

Software Heritage is a non profit multi-stakeholder initiative launched by Inria in partnership with UNESCO, hosted by the Inria Foundation, and with a growing number of partners. It is building the **universal archive and knowledge base of software source code**, at the service of society as a whole.

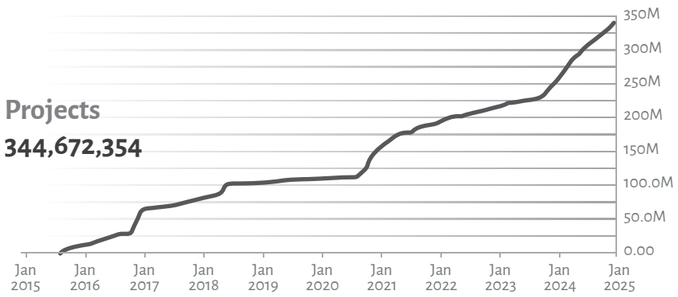
Source Files
22,548,654,226



Commits
4,743,656,085



Projects
344,672,354



Directories
17,750,848,771

Authors
85,859,826

Releases
101,802,909

Cultural Heritage

Industry

Research

Public Administration

Software Heritage

Learn about the first five years of Software Heritage in just five minutes

https://youtu.be/Ez4xKTKJO20

5 Advisors

17 Team members

1 Visiting hacker

31 Ambassadors

24 Sponsors

The Software Heritage Symposium and summit 2024 took place at UNESCO's headquarters on February 1st bringing together advisors, sponsors, ambassadors, and the entire community.
© Inria | M. Magnin

The **Software Heritage archive** is the largest collection of publicly available source code ever built. As of December 2024, it contains over **22 billion unique source files** from over **340 million software origins**.

Hosted by

RÉPUBLIQUE FRANÇAISE
Liberté Égalité Fraternité

Inria

Inria FONDATION

In collaboration with

unesco

Inria launched Software Heritage in 2015.

Diamond sponsors

Software is key in **CEA**'s commitment to transferring knowledge from research to industry.
With the Software Heritage Foundation, we stand behind the preservation and sharing of this knowledge.

Platinum sponsors

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE

The National Open Science Plan was launched on 4 July 2018 by the Minister of Higher Education, Research and Innovation. This plan includes a provision to support Software Heritage, an initiative that we consider a major pillar of open science. In addition to enabling open access to publications and research data, making research software source code openly available is critical to success of the open science program that we are collectively building.

CNRS's support to Software Heritage, a universal, open and sustainable software archive, is a natural part of our proactive approach in favour of open science, a necessary revolution in which everyone must play a part.

Microsoft has been involved in open source initiatives by enabling, integrating, releasing and contributing to many open source projects and communities for well over a decade. We applaud the Software Heritage as an open project that will help curate and conserve human knowledge in the form of code for future generations as well as help today's generations of developers find and re-use code worldwide.

Intel has been at the forefront of open source development for nearly two decades and today is a top contributor to the Linux kernel, as well as dozens of leading projects across technology markets and industries.
Intel is committed to support Software Heritage in its mission to collect, preserve and share code, as we believe open source is critical in transforming our world through innovation in enterprise, consumer technology, the Internet of Things and beyond.

Huawei has been working with the open source communities for decades: we are active contributors in projects ranging from the Linux kernel to cloud native computing and machine learning, and we will keep increasing our participation and investment in this open innovation world. We share Software Heritage's vision that publicly available source code, including open source software, is a precious heritage of mankind, and should be collected, preserved and shared for the benefit of all.

Gold sponsors

Google is proud to support Software Heritage in its mission to collect, preserve, and share software for future generations. We look forward to the variety of services that can be built atop this unique collection of software.

Partnering with Software Heritage was a great journey for BigCode and **Hugging Face**. The foundation's focus on preservation, reproducibility, availability and traceability mirrors many of the values and mission of Hugging Face as a central platform for sharing and collaborating in the ML community.

Open source software has been one of the instrumental, driving forces of innovation this century. Software Heritage is an important organization for software, (...). Archiving of code in a curated form maintains the technical and scientific knowledge that goes along with the code, preserving the innovation while also providing a means for determining prior art.

At **ServiceNow** we recognize the value and importance of preserving open-source software (...). We firmly believe in the capacity of Software Heritage to cultivate goodwill and collaboration within the technology ecosystem, while promoting a more sustainable and open software industry.

Firmly committed to open science, which is at the heart of its project, **Sorbonne University** supports Software Heritage. By helping to collect and to share software, Software Heritage contributes to one of the key missions of the university: the preservation and transmission of knowledge and of our scientific heritage.

By supporting the Software Heritage initiative, Université de Paris continues its commitment to the free and responsible sharing of knowledge and research software.

Silver sponsors

La DINUM

Bronze sponsors

Our mission

Our goal is to **gather**, **protect**, and **share** all **publicly available source code**. This foundation can support a vast array of applications, from cultural heritage and education to industry, science, and public administration, and more.

"Programming is the art of telling another human being what one wants the computer to do."

— Donald Knuth

Collect

Software is the lifeblood of our digital age. Every line of code, every software component, has the potential to shape our future. We're committed to preserving this digital heritage by archiving all publicly available source code and its complete development history. This rich metadata is carefully harvested and structured, enabling the creation of curated archives on top of Software Heritage.

Preserve

Software is fragile. We risk losing valuable technical and scientific knowledge when popular code-hosting platforms falter. To preserve this digital heritage, we archive software and its development history.

This monumental task requires a collective effort. To foster collaboration, we'll release all our software and technical documentation as open source. By sharing our tools and processes, we aim to empower others to contribute to this vital mission.

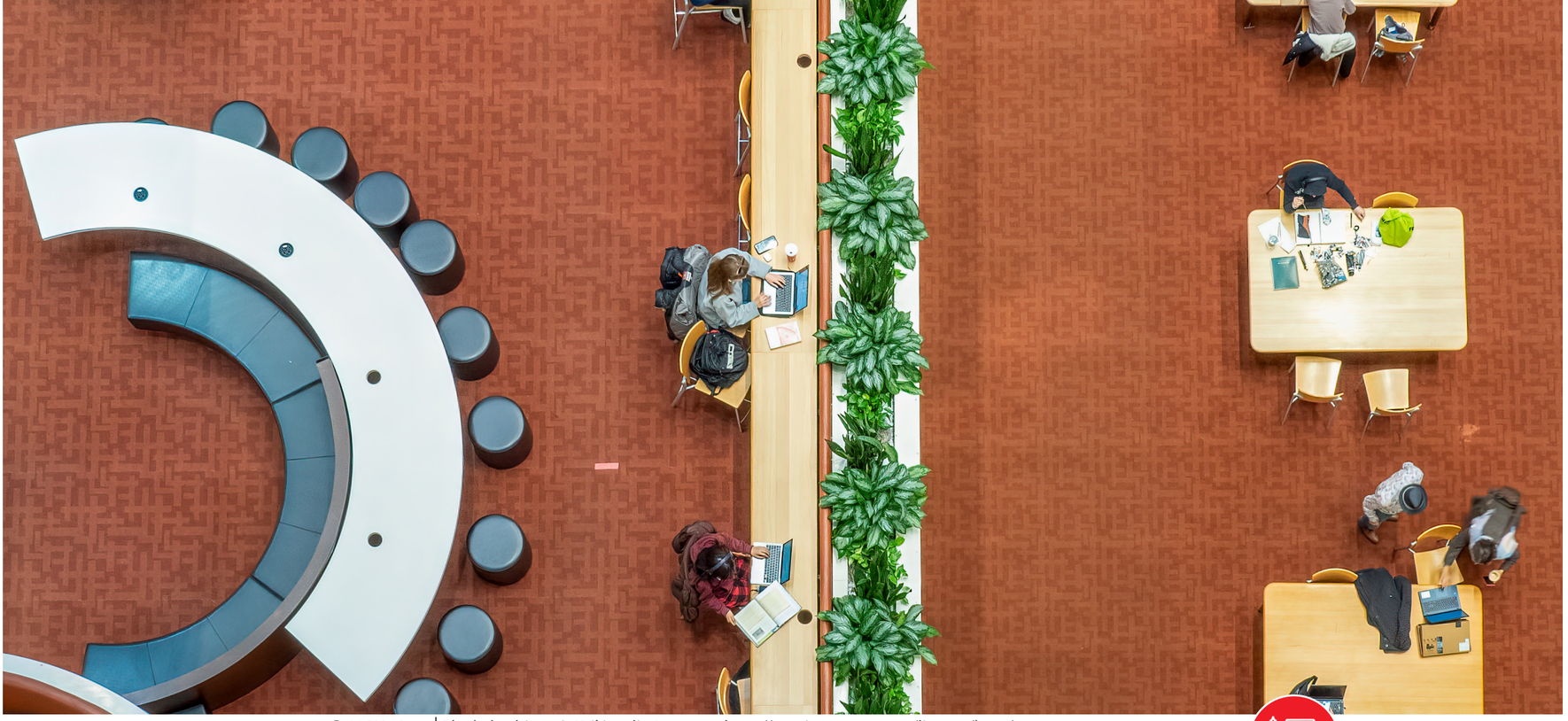
We're building a resilient network of peers and mirrors to safeguard multiple copies of the software we collect. This distributed approach ensures the long-term preservation of our digital heritage.

Share

We're constructing the world's largest archive of software source code. Our goal is to index, organize, and make this invaluable digital heritage accessible and maximally useful.

To ensure the long-term preservation of this knowledge, we assign unique Software Hash IDentifiers (SWHIDs) to each software component. These identifiers, independent of any central registry, form the foundation of a robust and resilient knowledge network.

We're developing a suite of services, from documentation and classification to advanced search and distribution, to realize the full capabilities of this digital library.



Software Heritage in a nutshell

We're building a critical infrastructure to ensure three key attributes for the source code we collect:

Availability

Long-term storage and preservation of code.
Accessible to researchers and developers.

Traceability

Unique SWHID identifiers for each software component.
Reliable and persistent references.

Uniformity

Consistent access to diverse source code.
Unified Application Programmer's Interface (API).

Software Heritage: Ethical charter for using archive data

Avoid Harm: Users must consider the potential ethical implications of their data usage. Avoid actions that could cause harm, even with well-intentioned research.

Protect Privacy: Uphold policies to protect personal data within the archive. Respect the privacy of individuals who contribute to the shared software commons.

Preserve data integrity: Discourage extensive redistribution of the Archive. Use persistent identifiers within Software Heritage to maintain data stability over time.

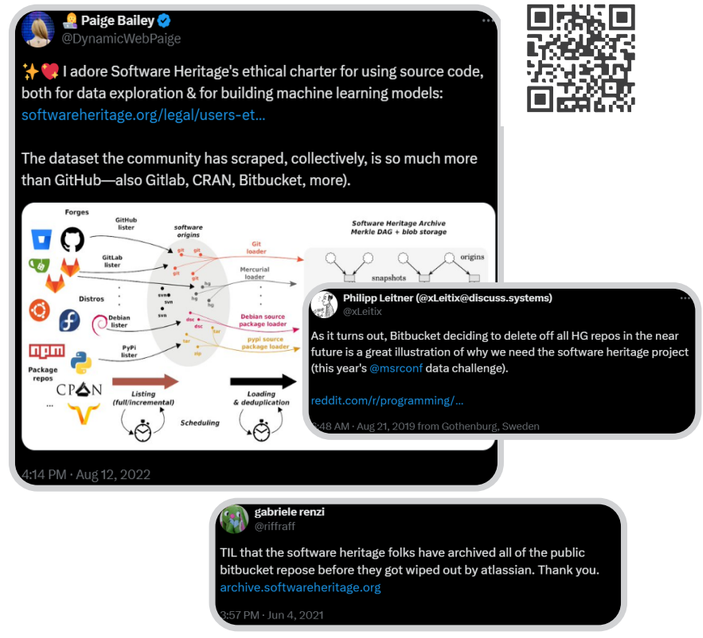
Responsible data usage: Users are responsible for ethically handling derived data from their analysis. Refrain from disseminating sensitive informationn.

A catalog to find them all

Software is spread all around: it is developed on many collaborative platforms and distributed through a variety of different channels. Software Heritage is building a **universal catalog** to let you **find** all software projects, no matter where they are developed, or how they are distributed.

An archive to preserve our digital heritage

Modern software development relies on collaborative platforms, many of which are free to use. These platforms allow users to create, modify, and delete projects, but they're not designed as long-term archives. In recent years, we've witnessed the sudden closure of several platforms, jeopardizing countless software projects. Software Heritage is building the universal archive needed to prevent the loss of source code.



An instrument to explore and study them

Software has become the backbone of our modern world, underpinning every aspect of our lives. In a few short decades, we've constructed incredibly complex software systems. Some are massive programs with tens of millions of lines of code, while others are smaller but rely on hundreds or thousands of other components.

To build better, safer software systems and protect against malicious attacks, we must master this complexity. Just as we've developed tools to explore the cosmos, it's time to create a shared infrastructure to explore and study the vast expanse of software development. With sufficient support, Software Heritage can evolve into this essential infrastructure.

Measuring the reach of CERN's code

Just as the European Organization for Nuclear Research (CERN) explores the fundamental particles of the universe, Software Heritage acts as a vast observatory for open-source software, cataloging its evolution and contributions across the globe.
Measuring the impact of open-source projects has always been a challenge. CERN is using Software Heritage to track their code, offering a more accurate picture of their contributions to the scientific community.

Tech highlights

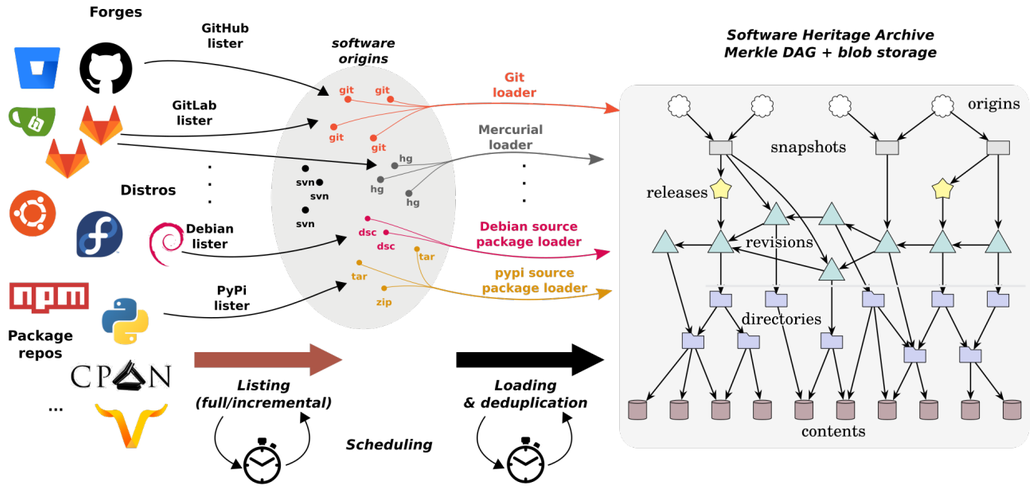
Public code history in a giant graph

Merkle graphs and SWHID

A massive crawler harvests source code from different sources and converts it, with all its development history, into a single giant Merkle directed acyclic graph, using SWHID cryptographic identifiers for all its nodes.

50 billion nodes | **700** billion edges

The Software Heritage data structure is a natural extension of Merkle trees, a classical cryptographic construction, combining a tree and a hash function. [Merkle, 1987]



The process involves three phases: identifying software sources, scheduling updates, and collecting software artifacts.

Software Heritage Loaders

A loader is a software component that ingests software artifacts into the Software Heritage archive, converting them into a Merkle graph. In 2022, we launched a dedicated page with all available loaders and links to their documentation.

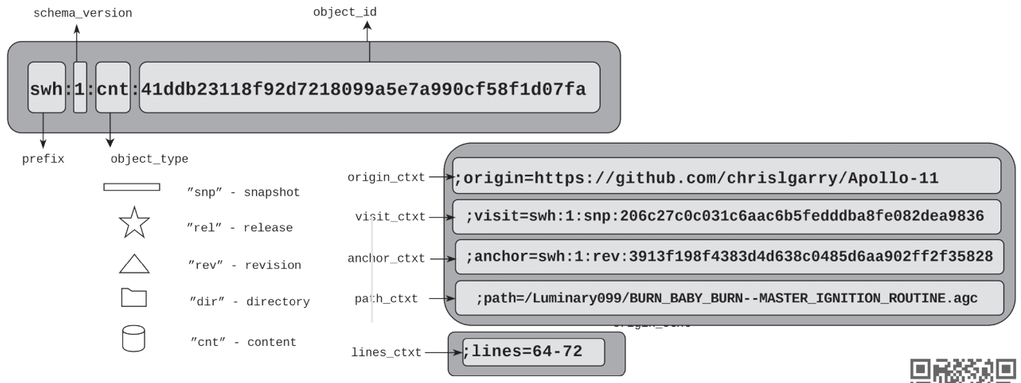
Software Heritage Listers

A lister is a software component that identifies software projects on code hosting and distribution platforms. All the available listers and links to their high-level documentation have been available since 2022 on a dedicated page.

The SWHID intrinsic persistent identifiers

All artefacts in the Software Heritage archive get a **SoftWare Hash IDentifier**, **SWHID** for short, guaranteed to remain stable and persistent over time.

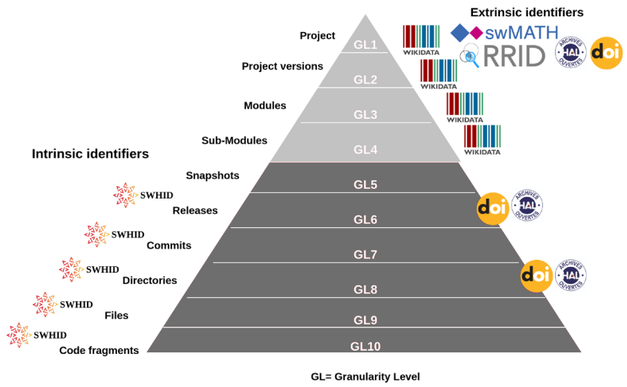
A SWHID consists of two parts, a mandatory *core identifier*, and an optional list of *qualifiers* that specify the context and can pinpoint a subpart. One can obtain them using the Permalinks sidebar present on all pages of the Software Heritage archive, and the core identifier can be computed independently by anyone.



Intrinsic and Extrinsic identifiers

Building a solid web of knowledge that lasts over time is of paramount importance. A key component of this web are the links between entities, that are designated using systems of identifiers in two broad categories:

- Extrinsic: use a register to keep the correspondence between the identifier and the object (e.g. URLs, DOIs)
- Intrinsic: intimately bound to the designated object, they do not need a register, only agreement on a standard (e.g. git cryptographic hashes)



The software development world has long adopted intrinsic digital identifiers, like Git hashes, that enable decentralized operations and independent integrity verification. What makes SWHIDs special is that they do not depend on a version control system: any software artifact ingested in the Software Heritage archive gets these identifiers.

All the levels of granularity that correspond to concrete software artifacts: snapshots, releases, commits, directories, files and code fragments.

SWHIDs are part of the SPDX 2.2 industry specification and have been adopted by Wikidata and various infrastructures.
Research Data Alliance/FORCE11 Software Source Code Identification WG, 2020, https://doi.org/10.15497/RDA00053

Collaborative work on the SWHID publicly available specification

The SWHID Working Group develops and maintains the Software Hash Identifier specification, fostering an open and collaborative environment for its evolution. Participation in the working group is inclusive, encouraging contributions from a diverse range of individuals via a team mailing list and regular meetings.

In November 2023, the working group released the SWHID Specification Version 1.1. In 2024, the SWHID specifications were submitted for standardization as ISO:IEC DIS 18670.

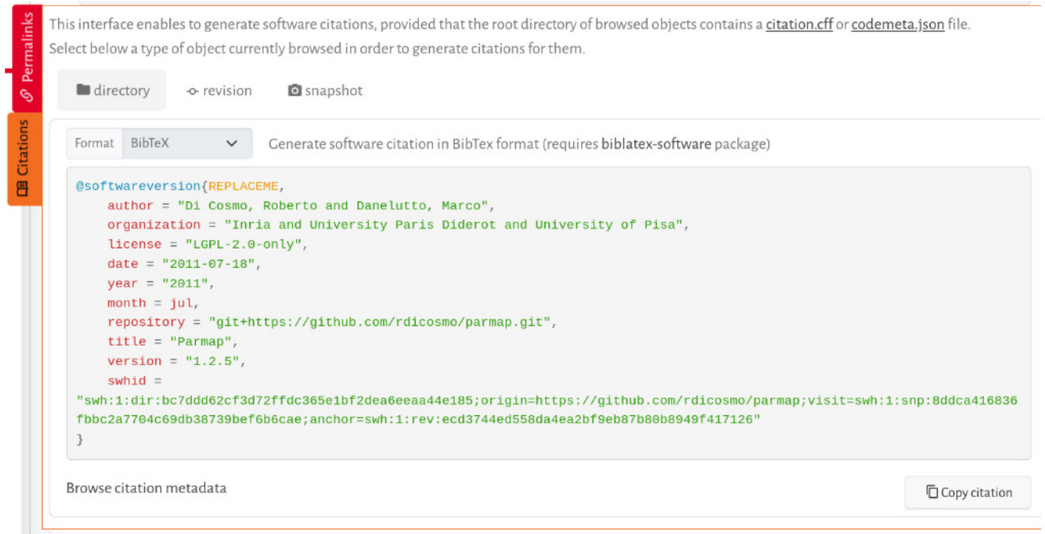
The precise identification of software artifacts and versions is crucial, supporting Software Heritage's mission to collect, preserve, and share source code. By using the SWHID for over 50 billion artifacts, Software Heritage ensures clear referencing and retrieval. The SWHID Working Group, through an open process, has developed and approved a comprehensive specification, which is now publicly available. The term "Software Hash Identifier" underscores its potential use cases beyond Software Heritage.

Citations

Now, various levels of granularity can be referenced using the SWHID, along with three bibliographic entries directly sourced from the Software Heritage archive, when a codemeta.json or a CITATION.cff file is available:

- **@software** Computer software
- **@softwareversion** A specific version of software
- **@codefragment** A code fragment (e.g. a specific algorithm in a program or library)

Biblatex-software is integrated in CTAN and TeXLive, and works out of the box in Overleaf. As of April 2022, included in ACMART Class for typesetting publications of ACM.



Screenshot: Citation feature on the Software Heritage archive for the Parmap software

SWHID Working Group

Software Heritage **Services**

Browse & search

The archive is the gateway to all captured source code and its entire development history.
With the browsable platform, it's possible to visualize all visits made to a given location of the code (collected from different forges, package managers and distros) and read the source code content captured.

SWHID provider & resolver

We provide a persistent identifier (PID) that can identify each and every source code artifact with integrity, called a SWHID. SWHIDs are intrinsic identifiers which are intimately bound to the designated object, they do not need a register, only an agreement on a standard to resolve them. The SWHID can also be used as a badge.

Download

The vault is the service in charge of reconstructing parts of the archive as self-contained bundles, that can then be imported locally. For instance in a Git repository.
With the vault, directories and revisions can be downloaded by users on the web platform or through the API.
Go to the download vault API reference https://docs.softwareheritage.org/devel/swh-vault/api.html

Save Code Now

It will take some time to get to every repository in the world, especially if these repositories keep on changing several times a day. This is why the "Save Code Now" service is provided, to give the possibility to notify SWH with a save request.

Deposit

S.W.O.R.D (Simple Web-Service Offering Repository Deposit) is an interoperability standard for digital file deposits. It allows a client (a repository, e.g. HAL) to submit software source archives and metadata to the archive. Metadata can be also submitted referencing a repository url (origin) or a SWHID.

Add Forge Now

In 2022, we introduced a feature called "Add Forge Now", to allow any user to propose archiving of *a whole forge*. The process follows a validation workflow, including curation and verification that the forge technology is supported by Software Heritage tools.

New in 2024

Bulk on-demand archival

A bulk version of the "Save Code Now" feature, which provides a dedicated pipeline for rapid ingestion of a large list of origins, and provides real-time status information for the ingestion progress.

Citation

A new web UI feature enables the generation of software citations, provided that the root directory of the browsed objects contains a citation.cff or codemeta.json file. The first supported format for generated citations is the BibTeX format.

swh-scanner

The code scanner is a SWHID-based command-line interface tool that compares a local codebase with SWH archive to identify which artifacts are already known in the archive, and retrieves information on possible **provenance** (origin of the first occurrence.)

Software Heritage **technical roadmap**

Want to see what's next?
The technical roadmap is online!

Browser extension

Read the documentation

Behind the scenes

API

API access is over HTTPS. All API endpoints are rooted at https://archive.softwareheritage.org/api/1/ and the data is sent and received as JSON by default. You can jump directly to the endpoint index , which lists all available API functionalities, or read on for more general information about the API.

GraphQL

GraphQL allows a client to fetch the server data using a query language and enables them to create powerful requests.

Architecture

Archiving a repository from a forge isn't the same as archiving source code from a package manager. It's even harder because version control systems have evolved a lot over the decades. Software Heritage architecture was designed to harmonize different sources into a robust infrastructure.

Metadata

SWH collects and extracts metadata that describes and provides additional information on source code.

- Extrinsic metadata are metadata which are not found in the software source code.
- Intrinsic metadata are metadata included in the source code, in a specific file or as part of a source code file.

Indexing

The swh-indexer extracts the following:

- mimetype ,
- ctags,
- language,
- fossology-license (detecting the license of a file),
- Intrinsic descriptive metadata which is found in metadata files in the source code (e.g package.json, codemeta.json, pom.xml).

Data model

The data model adopted by Software Heritage to represent the information that it collects is centered around the notion of software artifact, using the following canonical names, from bottom to top: contents, directories, revisions and releases. It also uses origins, visits and snapshots to store provenance information.

Provenance

Based on a cutting-edge approach of the SWH graph analysis to generate a provenance index, this API enables retrieval of the probable origin of the first occurrence for given content SWHID or a list of SWHIDs.

Webhooks

Webhooks are a powerful tool for automation.
In the realm of code hosting platforms, they're an essential bridge between events and actions triggering a predefined action whenever a particular event occurs in your repository.

©Inria | B. Fourrier

Culture and education

A shared infrastructure for multiple stakeholders

> *"[We call to] support efforts to gather and preserve the artifacts and narratives of the history of computing, while the earlier creators are still alive"*
> — Paris Call on Software Source Code[1]

Cultural heritage is the legacy of physical artifacts and intangible attributes of a group or society that are inherited from past generations, maintained in the present and bestowed for the benefit of future generations.
Software in source code form is produced by humans and is understandable by them; it is a special form of **knowledge** that is at the same time **human readable and machine executable**.
It is an important part of our heritage that we cannot afford to lose. Software is furthermore a key enabler for preserving other parts of our cultural heritage that we would de facto lose if we lose the software needed to access them.
Preserving software is essential for preserving our cultural heritage.
We have the privilege to be able to talk to most of the people that created this new science and technology of computing, but there we have little time left: it is **urgent** to take action, and Software Heritage is providing guidance and tools, in addition to the archive infrastructure itself.

1. UNESCO website ark:/48223/pf0000366715, 2019.

"Telling historical stories is the best way to teach. It's much easier to understand something if you know the threads it is connected to."

From: *"Let's Not Dumb Down the History of Computer Science."*
Donald E. Knuth, Len Shustek

Software Heritage acquisition process

A step-by-step video tutorial

The Software Heritage Acquisition Process (SWHAP), developed in collaboration with UNESCO and the University of Pisa, is now complemented by two video tutorials: an introductory video and a step-by-step guide.

Software Stories

Highlighting the human side behind the software projects

Software Stories is a project supported by UNESCO as part of the shared mission to collect, preserve and share source code as a precious asset of humankind. The Software Stories system allows users to create a multimedia overview of a landmark legacy software title, making it accessible to a wide range of software enthusiasts without any technical background.

© Personal archive | M. Fichen

SWHAP workshop

Following the Inria legacy software survey, a small group of former and current Inria employees was invited to participate in a preservation experiment. Volunteers were provided guidance, including an online Q&A session, to navigate the SWHAP process. To wrap up the initiative, a workshop was organized to gather feedback on the proposed processes and tools. During the workshop, participants also had the opportunity to experiment with creating Software Stories.

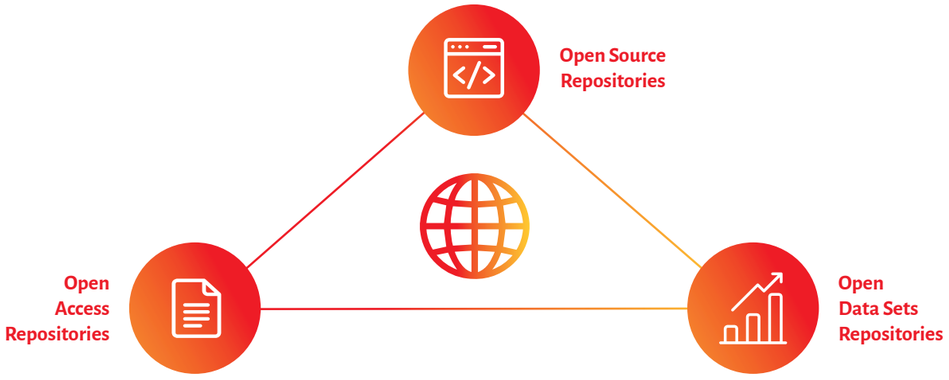
©Inria | 1968-001-001

Preserving Inria's software heritage

Software Heritage, in collaboration with the Inria alumni network and the Direction of Culture and Scientific Information (DCIS) of Inria, extended an invitation to former employees of Inria to contribute to the inventory of the software heritage built at Inria since its inception. This initiative resulted in the submission of over a hundred software items. Findings from the survey were presented at the iPres 2024 International Conference on Digital Preservation and led to a conference paper titled "Preserving Inria's Legacy Software: A Crowd-Sourced Approach."

The software pillar of Open Science

Software has become a pillar of research, ubiquitous in all its fields: a large part of the technical and scientific knowledge that is being developed today is described in the **software software code** at a level of detail that is often needed to remove ambiguities that may exist in intuitive descriptions. The preservation of this universal body of knowledge is as essential as preserving research articles and data sets. In the quest to make research results **reproducible**, and pass knowledge to future generations, we must preserve these **three main pillars**: research **articles** that describe the results, the **data** sets used or produced, and the **software source code** that embodies the logic of the data transformation.



The French National Software & Source Code College actively executes the second national plan for Open Science in France. Among its key missions is the commitment to ***"contribute to the production and dissemination of reference methodologies and good practices relating to the production and governance of projects, including with regard to their referencing, sustainability, enhancement and heritage preservation."***

French second national plan for Open Science, July 2021

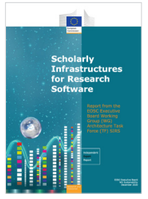
"Open Scientific knowledge [includes] open source software: source code must be included in the software release and made available on openly accessible repositories."

Unesco recommendation for open science, November 2021

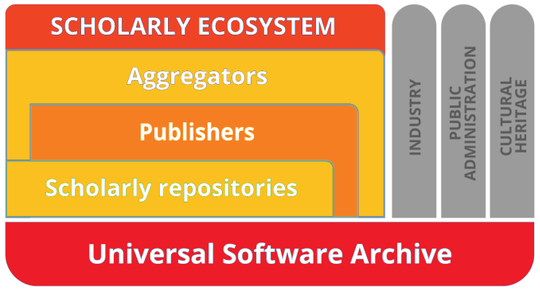
Serving the scholarly ecosystem

Software Heritage builds bridges between academia and the rest of the software world. Actively collecting all available source code online and fostering partnerships for software deposits. Helping researchers in citing and describing software, enabling reproducibility.

Archive, reference, describe and credit research software



Software source code is much more than data, it is a creation of the human ingenuity, and research software research software calls for specific solutions in scholarly infrastructures.



Scholarly repositories

Scholarly repositories and Software Heritage collaborate to support the archival, reference, description, and citation of research software by enabling the deposit of artifacts into Software Heritage, exposing SWHIDs in repository records, exchanging curated metadata, and exporting citation information in common open formats like BibLaTeX and codemeta.json.

The SWHID deposit for Research Software on HAL and Software Heritage is available on all HAL instances since January 2023. Thanks to a close collaboration between the CCSD, IES-INRIA and the Software Heritage team. The SWHID deposit is an addition to the already existing research software deposit as a compressed archive. Since 2024, software submitted to Zenodo is transferred to Software Heritage.

Publishers

IPOL, the Image Processing On Line journal, and **eLife**, both archive research software and deposit metadata in Software Heritage.

Aggregators

swMath maintains a collection of over 40000 mathematical softwares, and archives them in Software Heritage. OpenAire integrates information from Software Heritage in their knowledge graph.

European projects

Our goal, through participation in european projects, is to further recognition of the importance of software in research, building tools and services to interconnect with the scholarly ecosystem, and improve guidelines to foster their broad uptake.

eosc | FAIRCORE4EOSC
Core Components Supporting a FAIR EOSC

In the FAIRCORE4EOSC project, the Beta version of the Research Software APIs and connectors was released, linking research output infrastructures with the Software Heritage archive.

SoFAIR

The SoFAIR project enhances the findability and accessibility of research software by linking it to publications. Funded by CHISTERA, it aims to improve the discoverability and reusability of open research software, supporting accessibility of software source code.

eosc | FAIR-IMPACT
Expanding FAIR solutions across EOSC

FAIR-IMPACT European project supports and disseminates FAIR-enabling practices, tools and services across scientific communities at a European, national and international level. In this project. Software Heritage led the development of the **Research Software Metadata Guidelines** and contributed to the governance efforts around the CodeMeta initiative.

CodeMeta simplifies translation between vocabularies, extending Schema.org to represent software metadata in a consistent JSON format, codemeta.json.
https://github.com/FAIR-IMPACT/RSMD-guidelines

Adoption and **recognition**

Building strong relationships is a key part of Software Heritage's strategy. An infrastructure is more than just technology; it's about articulating values and providing services that meet the needs of our users.

Understanding user needs

To foster collaboration and identify use cases, Software Heritage actively engages with organizations like RDA, Force11, and CodeMeta. Additionally, we encourage direct collaboration between our team and the broader community.

Partnering in the FAIR-IMPACT Open Call

Software Heritage is a partner in the FAIR-IMPACT project to increase recognition of research software and improve software curation and metadata standardization in the scholarly ecosystem.

Presenting SWHIDs at PIDfest 2024

The Software Hash Identifier (SWHID) uniquely identifies software source code artifacts with intrinsic persistence, ensuring integrity and traceability has been presented during PIDfest 2024.

Software as infrastructure, and infrastructures for software

Software is essential for modern research in all fields, as shown by precise factual evidence, and is as such a fundamental infrastructure for research.

"Towards Preserving Digital Culture," the interviews series by Camille Françoise

Software Heritage ambassador Camille Françoise asked experts to share their thoughts on software preservation in an interview series published on Medium.

2024 Software Heritage community workshop

This gathering, brought together developers, curators, librarians, open source advocates, and researchers from around the world to share ideas and collaborate toward a common goal.

Easing adoption

To drive cultural change and accelerate adoption, Software Heritage is committed to reducing barriers and building a supportive community. We're empowering advocates to serve as "cultural brokers," bridging the gap between the platform and its users. We're also actively disseminating knowledge in diverse learning environments to reach users where they are.

Course: Reproducible Research II: Practices and tools for managing computations and data

Course contents are aimed at researchers and engineers who are looking for good control of their computing environments. software source code artifacts with intrinsic persistence, ensuring integrity and traceability has been presented during PIDfest 2024.

Lesson: Preserve and identify research software

This lesson in French introduces archiving methods tailored to the specificities of software and various user profiles. It covers best practices for citing software and methods for effectively exploring the Software Heritage collections.

The EOSC Opportunity Area OA7 experts group: Research Software

The primary objective of this Expert Group is to address the challenges and opportunities around research software in the context of the EOSC framework.

Moving from awareness to uptake

SciCodes consortium

Software Heritage is part of the SciCodes consortium for scientific software registries and repositories.

Replicability stamp

Since 2023, the Computer Graphic Replicability Stamp Initiative (GRSI) platform has adopted Software Heritage to archive and reference research software.

Running software, again and again

Software Heritage ensures the availability and traceability of source code, essential for software reproducibility and reuse. Guix, an advanced GNU distribution focused on reproducibility, aim to connect Software Heritage with package managers and build systems to replicate complete executables and systems.

French research strategy

Software Heritage has been recognized in France's national research infrastructure strategy, alongside the HAL portal, for its role in archiving, referencing, and citing research software. The national research funding agency recommends it for all funded projects.

Funding agencies recommendations ANR 2023 guidelines

Membership for Open Science
Archives and Libraries Interest Group (ALIG)

Libraries advance teaching, research, and learning by providing resources, enabling discovery, and offering expert guidance. As software source code becomes increasingly central to contemporary scholarship, libraries must support researchers who work with it.
The Archives and Libraries Interest Group (ALIG) brings together stakeholders that are interested in supporting unfettered access, reference and citation of software produced by academic research, reinforcing the principles of open science. By joining the ALIG, you become part of a community of libraries that not only utilize the archive but also actively contribute by sharing your specific needs. You can exchange experiences and best practices, and help shape new features. Your input is essential for Software Heritage to adapt and evolve according to the diverse needs of its users.

© Unsplash | Quino Al

How libraries shape the future of research infrastructure

Libraries advance teaching, research, and learning by providing resources, enabling discovery, and offering expert guidance. As software source code becomes increasingly central to contemporary scholarship, libraries must support researchers who work with it.

In this series of interviews, professionals share their approach to research software.

Canadian Research Knowledge Network
Réseau canadien de documentation pour la recherche

CRKN

Memorial University of Newfoundland,
Université de Montréal,
University of Toronto,
York University,
University of Saskatchewan,
University of New Brunswick,
Western University.

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

Consortium of Swiss Academic Libraries (CSAL)

Universität Bern
ETH Zürich
Universität Luzern
Université de Neuchâtel
CERN

© iStock.com | ktsimage

Supporting the supporters

Librarians are at the forefront of supporting researchers and contributing to institutional open science strategies. Software Heritage partners with these professionals by providing accessible documentation and training. Our participation in the 2024 ADBU congress helped build relationships and strengthen collaboration.

CCSD and Software Heritage shared a booth at the ADBU Congress, offering brief presentations on various services. Sessions covered the Ambassadors network, software deposit in HAL, and the SWHID.

SCOSS SUSTAINING THE OPEN

Joining the SCOSS family

The SCOSS board selected Software Heritage for its 5th pledging round in November 2023, recognizing its role in ensuring global access to research software. SCOSS members, including libraries and research funders, can support by pledging annual donations for three years, providing financial stability and access to the Software Heritage ALIG. Details are available on the SCOSS website.

Research on the largest **public archive of software source code**

Software Heritage datasets

Software Heritage curates and releases unique large-scale datasets about public code development, suitable for open, reproducible science in various research fields.
Some of the most recent and notable datasets include:

Software Heritage Graph dataset

Fully deduplicated Merkle DAG representation of the Software Heritage Archive. This dataset links source code files and directories, commit evolution over time as observed by Software Heritage during archival, providing a unique integrated view of the development of public code.

```
SELECT COUNT(*) AS C, word FROM (
   SELECT word_stem(lower(split_part(
    trim(from_utf8(message)),' ', 1)))
   AS word FROM revision
   WHERE length(message) < 1000000)
WHERE word != ''
GROUP BY word
ORDER BY C
DESC LIMIT 20;
```

QUERY

RESULTS

Results (20) — Copy — Search rows

#	c	word
1	271573294	updat
2	163328012	merg
3	140044381	add
4	105800317	fix
5	103646653	ad
6	52891401	bump
7	50067041	initi
8	45609622	creat
9	42633225	remov
10	32230842	chang

Antoine Pietri, Diomidis Spinellis, Stefano Zacchiroli
The Software Heritage Graph Dataset: Public software development under one roof
In the proceedings of MSR 2019.
Award: selected as the topic for the MSR Mining Challenge 2020

The Software Heritage licence dataset

6.9 million unique full texts of free and open-source license texts, extracted from the Software Heritage archive, with origin information and a ground truth to train machine learning tools.
Award: Data and Tool Showcase Award, MSR 2022.

Barahona, Montes-Leon, Robles, Zacchiroli
The Software Heritage License Dataset
(2022 Edition)
Empir. Softw. Eng. 28(5): 107 (2023)

Popular content filenames dataset

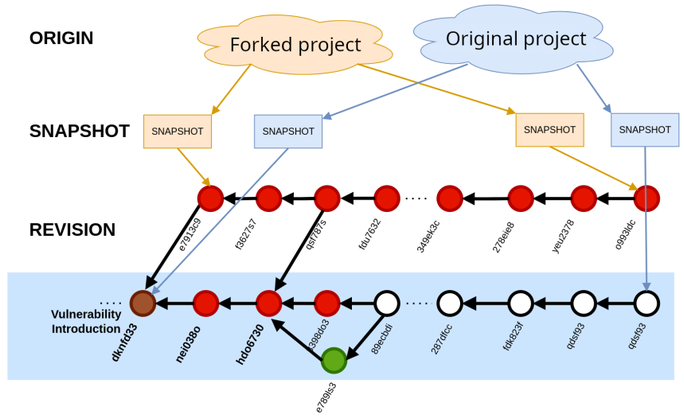
Provides the most common filename for each unique file content in the Software Heritage Graph dataset. This dataset simplifies the selection of specific file content subsets based on filename patterns, aiding research in areas like data compression and machine learning.

Valentin Lorentz, Roberto Di Cosmo, Stefano Zacchiroli.
The Popular Content Filenames Dataset: Deriving Most Likely Filenames from the Software Heritage Archive.
Data paper. July 2023

SWHSec: Leveraging Software Heritage to **enhance cybersecurity**

Free and Open Source Software (FOSS) constitutes 70-90% of any piece of modern software solutions." — (OpenSSF, 2022)
Securing FOSS is of paramount importance for securing the global software supply-chain. With its global view on the public development of FOSS and vast open knowledge base about it, Software Heritage is uniquely positioned to help secure open-source software to the benefit of all.

The SWHSec project, started in 2024, brings together partners from five research institutions to leverage Sofware Heritage to enhance cybersecurity in several ways, from vulnerability propagation to static analysis, anomaly detection in repositories, and supply-chain hardening.
In the project's first year, we developed the capability to identify vulnerabilities (CVEs) in third-party forks of open-source projects. This allows us to detect forks created after a vulnerability was introduced but has failed to incorporate the corresponding security patches. Software Heritage can accomplish this independently of version numbers and across multiple development platforms, such as GitHub and GitLab. This approach surpasses existing state-of-the-art methods.

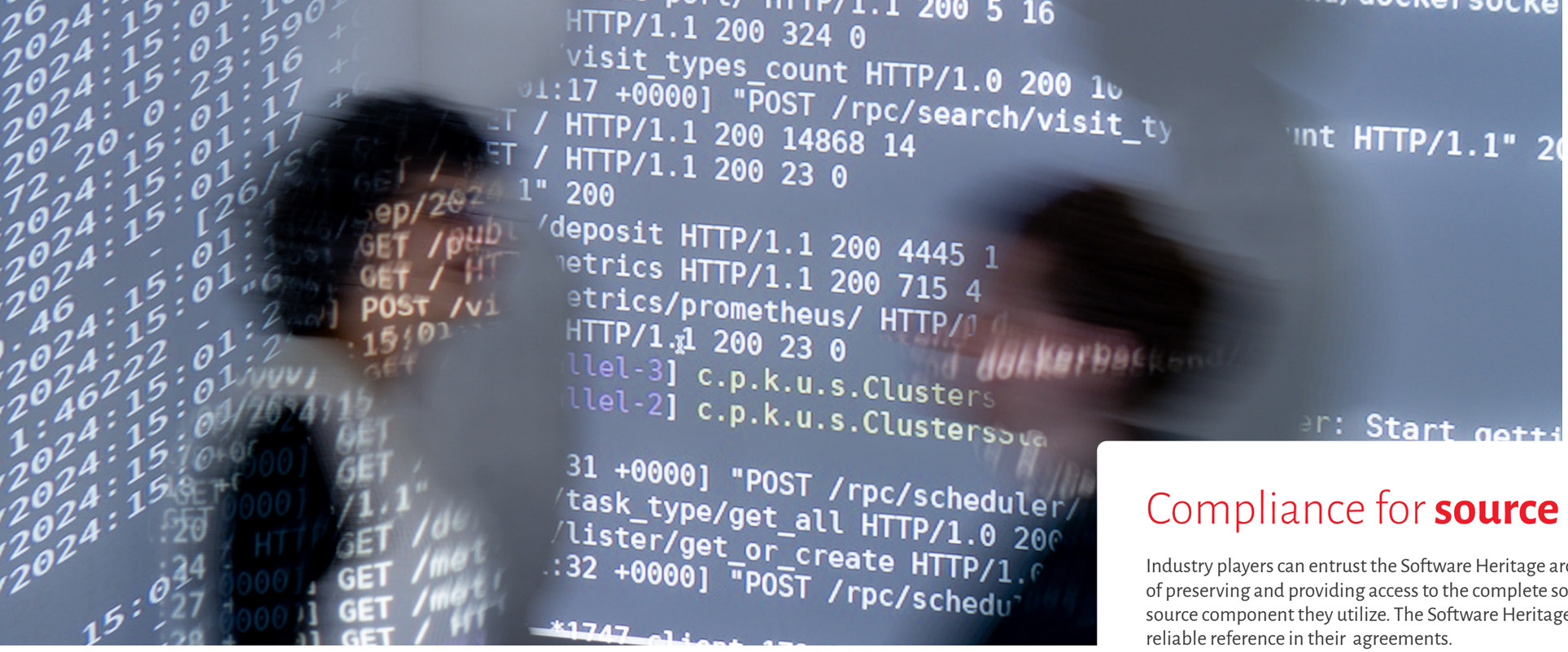


Lefeuvre, Reux, Zacchiroli, Barais, Combemale
Cross-fork and Cross-forge Vulnerability Propagation.
(Forthcoming)

Diversity, equity, inclusion in public code

Metadata in the archive can be used to study long-term trends of diversity in software development contributions. For example, male authors contributed 92% of public code commits up to 2019.
The ratio of female authors (and their contributions) has grown stably for 15 years reaching for the first time 10% of yearly contributions in 2019. The COVID-19 pandemic reversed the trend, inhibiting women's ability to contribute.

Casanueva, Rossi, Zacchiroli, Zimmermann.
The Impact of the COVID-19 Pandemic on Women's Contribution to Public Code
Emp. Softw. Eng. 30(1): 25
2025



©Inria | B. Fourrier

Industry

Contributing to security regulation implementation

The Cyber Resilience Act (CRA) will change the way industry handles security for open-source components. Software Heritage is a founding member of the Open Regulatory Compliance Working Group (ORCWG), hosted by the Eclipse Foundation. The ORCWG is dedicated to helping open-source projects navigate emerging regulatory requirements, such as the EU AI Act while ensuring that innovation and collaboration flourish. Software Heritage is an active member of the ORCWG's Steering Committee.

Securing the open-source software supply chain

The uniform, technology-neutral global Merkle graph, combined with the expanding mirror network, offers a clear and transparent source of trust. At all levels, source-code artifacts are tracked using the Software Hash Identifier (SWHID), which provides uniform, technology-independent, and cryptographically strong identification. SWHID normalization is ongoing.

Enhancing cybersecurity through Software Heritage

We bring together eight expert research teams specializing in security and software engineering to harness the power of Software Heritage's robust infrastructure and create cutting-edge tools for cybersecurity.

Tracking leaked code

The complexity of the software supply chain can sometimes result in code being inadvertently exposed that should remain private. Software Heritage is the only infrastructure that can track it across all the code hosting and distribution platforms globally.

Compliance for **source code distribution**

Industry players can entrust the Software Heritage archive with the responsibility of preserving and providing access to the complete source code of any open-source component they utilize. The Software Heritage archive can also serve as a reliable reference in their agreements.

Software Heritage is a long-term archive for source code

A non-profit, multi-stakeholder organization dedicated to collecting, preserving, and making the source code of all available software available forever.

Simplifies source code distribution

Companies can outsource their source code distribution obligations. Software Heritage acts as a trusted teward for long-term preservation.

Provides perpetual identifiers

Software Heritage provides SWHIDs—uniform, cryptographic identifiers for tens of billions of software artifacts. These allow anyone to independently verify the integrity of software without relying on a third party.

Integrates compliance processes

Source code deposits can be integrated into compliance workflows, saving time and resources. Unlike in-house approaches, Software Heritage's infrastructure reduces the need for dedicated resources over the long term.

Leverages shared infrastructure

By joining Software Heritage, you gain access to a globally supported, open infrastructure that brings together stakeholders from industry, open science, and digital preservation.

How it works

- **Prepare your archive**
  Create a complete and corresponding source code (CCS) archive.
- **Deposit your archive**
  Seamlessly integrate Software Heritage's API into your continuous delivery process to deposit the archive
- **Receive your SWHID**
  Obtain a unique, perpetual identifier (SWHID) for your archive.
- **Secure your code's future**
  Software Heritage will preserve your archive indefinitely.

This seamless process is fully compliant with all major copyleft licenses.

Contact us at **compliance@softwareheritage.org**

Empowering **public administration**

Promoting transparency and efficiency in the digital age

By sharing and reusing software, public administrations can enhance transparency and improve the services they provide to citizens.

Transparency and long-term preservation

Software Heritage offers a comprehensive archive for public software, guaranteeing its accessibility and preservation. By depositing software in this central repository, we ensure its availability for future generations.
The French Interministerial Directorate for Digital Services (Dinum) uses Software Heritage to systematically archive the open-source software developed and used by national public administrations. This strategic partnership contributes to the long-term preservation of public software assets.

https://code.gouv.fr/sources/#/repos

Depositing and sharing metadata

Public administrations can deposit machine-readable and structured metadata in the Archive, enabling sharing and reuse of information.

Towards a **global infrastructure**

Mirrors

Any data infrastructure faces multiple challenges over time, that can be technical, organizational or legal.
To minimize the risks over the long term, we are working to build a resilient system. Due to the nature of the archive, we follow a **centralized and replicated** approach, establishing a network of independent full **mirrors** of the archive.

In order to prevent information loss, and simplify access to humankind's software heritage, we are building an international network of ***mirrors***.
A ***mirror*** is a full copy of the Software Heritage universal source code archive, operated ***in agreement with***, but ***independently from*** the Software Heritage organization.

ENEA has launched the inaugural Software Heritage mirror, making it accessible to the public on December 13, 2023.

We look forward to see a variety of institutions from all around the world becoming progressively part of the mirror program.

Mirror ethical charter

"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

ENEA
Italian National Agency for New Technologies, Energy and Sustainable Economic Development

ENEA hosts the first Software Heritage mirror, and GRnet will also be hosting one. Several additional mirrors are currently under consideration.

Building for the long term

Lasting infrastructure

To ensure the long-term preservation of software, we have established a non-profit foundation and partnered with UNESCO. We're committed to open-source principles and transparent development practices. By sharing our code, specifications, and development processes, we aim to foster a vibrant and sustainable community dedicated to safeguarding our digital heritage.

Openness and transparency

We believe in open source and transparent development. Our code, specifications, and development processes are publicly available.

A collaborative effort

To ensure the long-term preservation of software, we foster a collaborative community. By working together, we can build a robust and sustainable infrastructure.

Facts and provenance

We follow best practices to store detailed provenance information, ensuring the accuracy and reliability of archived software.

Towards transparency in **artificial intelligence**

Empowering responsible AI development

Machine learning models promise to democratize the software creation process, making it easier for more people to benefit from the digital revolution.However, this transformative power hinges on transparency, traceability, and integrity.

Software Heritage gives you the tools you need to make this happen. We already archive over 22 billion files and 340 million projects, ensuring they're available for a long time and making it easy to add new code. Moreover, our SWHID system lets users reference over 50 billion software artifacts with unparalleled transparency and minimal cost. This robust foundation empowers researchers, developers, and organizations to build AI models responsibly and ethically while fostering a more inclusive and sustainable software ecosystem.

Software Heritage offers transparency, traceability, and integrity tools to help researchers, developers, and organizations build AI models transparently.

- Add new code easily. We already archive over 22 billion files and 340 million projects, ensuring long-term availability.
- Reference over 50 billion software artifacts with SWHID, while ensuring transparency at a minimum cost.

In 2024, Software Heritage launched the Statement on Large Language Models (LLMs) for Code, establishing three fundamental principles for responsible AI.

- Knowledge derived from the Software Heritage archive must be given back to humanity, rather than monopolized for private gain. The resulting machine learning models must be made available under a suitable open license, with the documentation and tooling needed to use them.
- All training data extracted from the Software Heritage archive must be clearly identified using SWHIDs to ensure transparency. Since all data in the archive is publicly available, researchers can easily access and analyze the training data. This traceability allows for the study of biases, verification of code origins, and proper attribution when generated code resembles training examples.
- Mechanisms should be established, where possible, for authors to exclude their archived code from the training inputs before model training begins.

In February 2024, BigCode released StarCoder2, trained on a transparent subset of GitHub-hosted repositories archived by Software Heritage, called The Stack v2. The project demonstrates that top-tier open models can be built while respecting these principles.

Challenges in AI training with public code

Training AI models with public code often involves redundant data downloads and inconsistent results. Addressing this requires a solution that eliminates the need for each team to clean and deduplicate data independently.

Software Heritage's deduplicated Merkle graph provides such a solution, streamlining the process and improving efficiency. At scale, license identification and code quality checks are nearly impossible, introducing legal and ethical concerns. Many datasets lack transparency, making it difficult to assess bias or legality; the Stack v2 dataset is a notable exception. Finally, the lack of reliable tools for tracing AI-generated outputs back to their training data raises accountability concerns.

The next frontier

Building a **Code Commons**

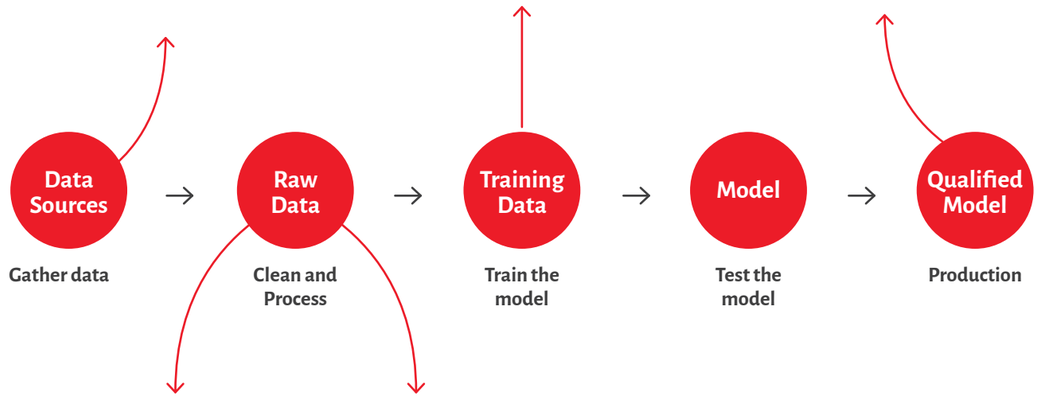
The CodeCommons project, funded by France 2030, brings together teams from Inria, CEA, Tweag, Scuola Superiore Sant'Anna, the universities of Bologna, Pisa, and Torino to address these challenges and pave the way for efficient, responsible, transparent and ethical AI on code.

Visit at: codecommons.org

Availability: Easy access to all relevant data for software (e.g., source code, pull requests, issues, discussions, etc.) with shared harvesting—done only once!

Traceability: Identify and make available the data used for training.

Ethics: Provide tools to verify the provenance and attribution of generative AI outputs.



Structuring: Organize and connect the various data sources to create a coherent training set.

Efficiency: Facilitate the extraction of qualified datasets to build high-performance models.

Supported by

© Inria | M. Gruenpeter. Software Heritage community workshop

Collaboration and community

"Alone we can do so little; together we can do so much."
- Helen Keller

A thriving community is essential to the long-term success of Software Heritage. That's why we partner with global funders to offer grants to experts who share our commitment to preserving software heritage for future generations.

Alfred P. Sloan Foundation

Thanks to a grant from the Alfred P. Sloan Foundation, Software Heritage has fostered a community of expert contributors to expand the Software Heritage archive. Seven subgrants have led to the archiving of over 300,000 new repositories.

NGI Zero

Four cascading NLNet Foundation grants enabled Software Heritage to rescue 250,000 endangered Bitbucket repositories, enhance the Mercurial loader, develop connectors for Nix and Guix, and experiment with the IPFS distributed file system.

NGI Search

This European project supports entrepreneurs, tech enthusiasts, developers, and socially-minded individuals who are reimagining how we search and discover information and resources online.

Get **involved**

The Software Heritage archive is dedicated to serving a diverse range of users, including cultural institutions, scientists, and industries. We invite everyone to join us in achieving these ambitious goals, here's how you can help.

Become a sponsor

Software Heritage's ambitious roadmap requires significant resources. We invite companies, institutions, and individuals to join our sponsorship program and support our mission.

Tackle scientific challenges

Building and maintaining a universal source code archive presents significant scientific challenges. We invite researchers to join us in addressing these challenges and contributing to our ongoing research.

Code with us

All of our software is open source. We invite you to contribute to our open-source projects and help us build the essential components to drive the archive forward. Your contributions will directly impact the future of Software Heritage.

Become a user

Find the tools and features you need to navigate Software Heritage. Connect, share, and collaborate with our community to build the future of the universal source code archive.

Grantees

©Inria | B. Fourrier

Meet our **team**

Executives

Roberto Di Cosmo *(Founder, CEO)*
Laetitia Cruse *(CFO)*
Stefano Zacchiroli *(Founder, CTO)*

Advisors

Serge Abiteboul *(French Academy of Science)*
Jean-François Abramatic *(EIT)*
François Bancilhon *(Data Publica / C-Radar)*
Gérard Berry *(French Academy of Science)*
Julia Lawall *(Inria)*

Management

Benoît Chauvet *(Project manager)*
David Douard *(Dev team manager)*
Morane Gruenpeter *(Head of Open Science)*
Nicole Martinelli *(Strategic communications)*
Vincent Sellier *(Sysadmin team manager)*

Engineers

Renaud Boyer
Nicolas Dandrimont
Antoine R. Dumont
Antoine Lambert
Valentin Lorentz
Guillaume Samson
Aymeric Varasse

Outreach

Sabrina Granger *(Open science community)*
Marla da Silva *(Communications and events)*

Visiting scientists

Mathilde Fichen

Visiting hackers

Hélène Jonin

Interns

Adèle Desmazières
Karim Ourdedine

In memoriam

Software Heritage mourns the loss of Lunar, aka Jérémy Bobbio, a valued member of our engineering team whose innovative spirit profoundly impacted our organization. Lunar passed away peacefully on November 8, 2024, in Rennes.
For memories and tributes from the team, visit: www.softwareheritage.org/2024/11/15/remembering-lunar

Ambassadors

At Software Heritage, we understand that success is achievable only through the collective efforts of a diverse community. Since 2020, our ambassador program has played a crucial role in nurturing collaboration and promoting adoption. This year, we added two new advocates with a wide range of geographical and technical expertise.


Agustín Benito Bethencourt


Alexis Lebis


Anna-Lena Lamprecht


Baptiste Mélès


Barış Güngör


Bertrand Néron


Bostjan Spetic


Bruno Khélifi


Camille Françoise


Cécile Arènes


Flavia Marzano


Frédéric Santos


Gavin Henry


Giacomo Lorenzetti


Harish Pillay


Italo Vignoli


Jaime Arias


Joenio Marques Da Costa


Julien Caugant


Linda Angulo-Lopez


Malin Sandström


Max Kalik


Maxence Azzouz-Thuderoz


Mohammad Akhlaghi


Océane Valencia


Pierre Poulain


Sandrine Layrisse


Shiraz Malla Mohamad


Simon Phipps


Violaine Louvet


Wendy Hagenmaier

Becoming an Ambassador

Interested in becoming a Software Heritage ambassador?
Tell us about yourself and your interest in our mission.

ambassadorprogram@softwareheritage.org

Software Heritage provides solid, common foundations to serve the diverse needs of heritage preservation, science, and industry.

softwareheritage.org
@swheritage
@SwHeritage
@swheritage@mstdn.social
@softwareheritage4978