Software Heritage

Annual Report 2023

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

Collecting, preserving and sharing software source code since 2015

Software is a precious part of our cultural heritage

Software is a precious part of our cultural heritage. We preserve and make accessible all the software we collect, because only by sharing it we can guarantee its preservation in the very long term.

Foreword

As we stand at the threshold of another year of innovation and collaboration, it's a privilege to reflect on the journey of Software Heritage. In 2023, we continued working on our long-term foundational mission, with more than 17 billion unique source files from over 270 million projects now safely archived. The "add forge now" functionality, now fully in production, has enabled the complete archiving of over 200 new forges.

A landmark achievement of 2023 was the inauguration of the first international mirror of Software Heritage at ENEA. This milestone marks the completion of a journey that began in 2019 and paves the way for establishing a mirror network that will place the vast expanse of humankind's software heritage beyond the reach of accident.

A key component of Software Heritage is the SWHID, an intrinsic identifier that is a cornerstone in ensuring the long-term traceability and reliability of all artifacts stored in the archive. This year marked a significant milestone in a global effort to make it widely available and usable: the SWHID, now referred to as the "software hash identifier", has been precisely described in an open specification produced by a broad working group.

Our community has always been the driving force behind our mission. This year, we count more than 30 ambassadors who have been instrumental in championing our vision worldwide. Their dedication and enthusiasm have been pivotal in broadening our impact and fostering a more inclusive and connected community.

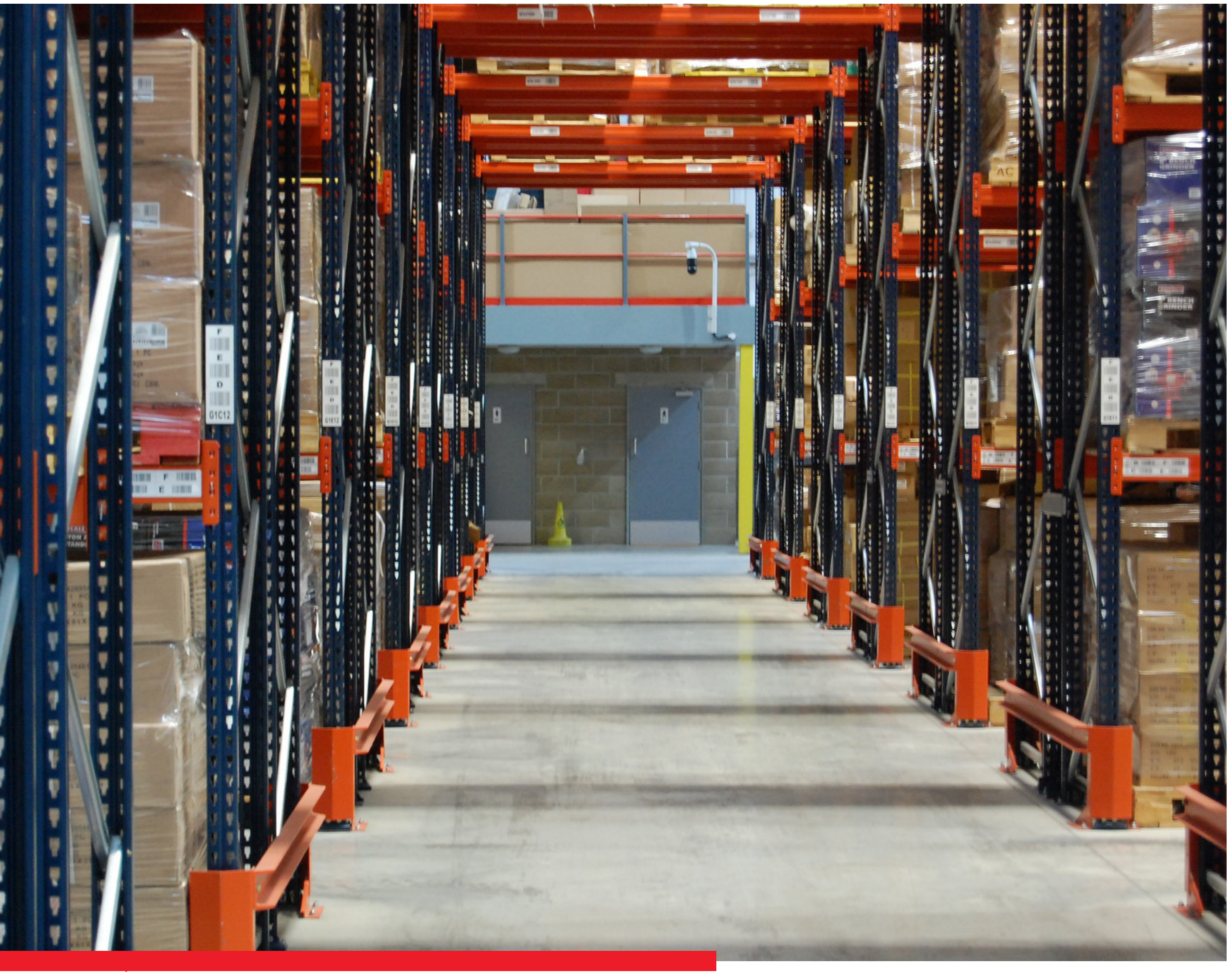
A significant endorsement has been the selection as a key infrastructure for Open Science by the Sustainability Coalition for Open Science Services (SCOSS). This acknowledgment is not just an honor, but a responsibility that we embrace wholeheartedly. It aligns perfectly with our vision of making software source code an accessible and integral part of scientific and technological discourse.

2023 also marked our foray into a new domain with the publication of our statement on large language models for code. We acknowledge the vast potential that the archive offers in this realm and are committed to making it beneficial for all of humankind.

As we look ahead, our resolve is stronger than ever. We are not just preserving source code; we are safeguarding a digital legacy for future generations. Our journey is made possible by the unwavering support of our members and sponsors, the collaboration of our partners, and the enthusiasm of our community.

We invite you to join us in this ongoing adventure, to contribute, to use, and to spread the word about our mission. With each passing year, our collective effort becomes more crucial, and our impact more profound.

Roberto Di Cosmo
Co-founder & CEO
Software Heritage



Sealey Warehouse by Mark Hunter, license: CC BY 2.0
https://www.flickr.com/photos/toolstop/4324416999/

WE HARVEST PUBLICLY AVAILABLE SOURCE CODE FROM MANY SOFTWARE PROJECTS AND KEEP UP WITH DEVELOPMENT HAPPENING THERE. AS OF TODAY OUR ARCHIVE ALREADY CONTAINS AND KEEPS SAFE FOR YOU:

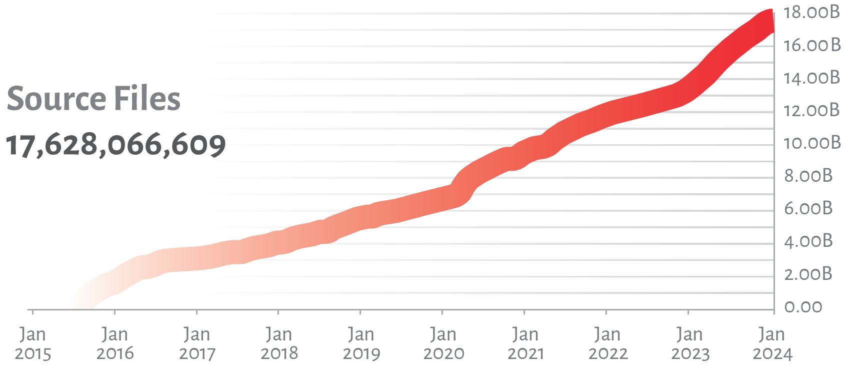
17,628,066,609	Source files
3,746,371,885	Commits
274,656,598	Projects

TABLE OF CONTENTS

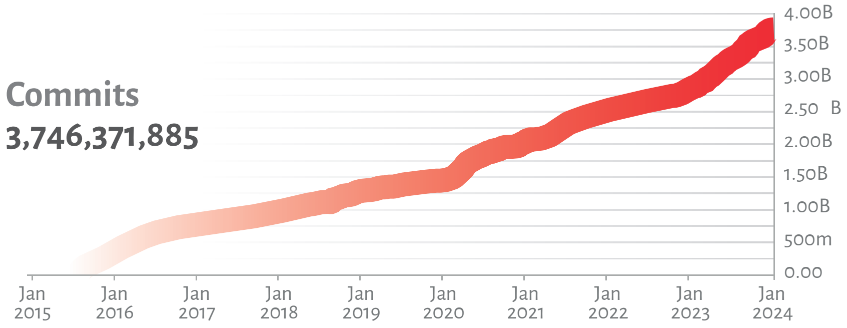
About Us

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.

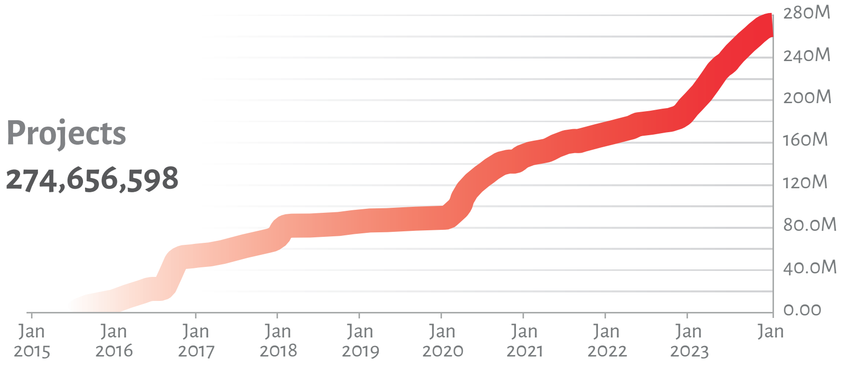
Source Files
17,628,066,609



Commits
3,746,371,885



Projects
274,656,598



Directories	Authors	Releases
14,171,783,106	**69,063,036**	**81,409,168**

CULTURAL HERITAGE
INDUSTRY
RESEARCH
PUBLIC ADMINISTRATION
Software Heritage

LEARN ABOUT THE FIRST FIVE YEARS OF SOFTWARE HERITAGE IN JUST FIVE MINUTES!

https://youtu.be/Ez4xKTKJO2o

4	17	3	33	22
Advisors	Team members	Visiting hackers	Ambassadors	Sponsors worldwide

The Software Heritage Symposium and summit 2023 took place at UNESCO's headquarters on February 7th bringing together advisors, sponsors, ambassadors, and the entire community.

The **Software Heritage archive** is the largest collection of publicly available source code ever built, containing, as of December 2023, over 17 billion unique source files from over 274 million software origins.

Hosted by

Inria
Inria La Fondation

In collaboration with

unesco

Software Heritage has been launched by Inria in 2015.

Sponsors

Diamond Sponsors

cea

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.

Gold Sponsors

Hugging Face

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**invention***network*

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.

servicenow

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.

SORBONNE UNIVERSITÉ

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.

Université Paris Cité

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

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

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.

SOCIETE GENERALE

We are aware of the code's value for our digital transformation, it has become a major asset for the bank and we firmly believe that we must preserve it in the long term. Open Source lies at the heart of our strategy, as it is in line with our needs and our values: team spirit, innovation, responsibility and commitment to better serve our clients.

Microsoft

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

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

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.

Silver Sponsors

AdaCore

Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE

GitHub

Google

UNIVERSITÀ DI PISA

Bronze Sponsors

Red Hat

SCANOSS

SCUOLA NORMALE SUPERIORE

Stuttgart Public Library, Germany by O Palsson, license: CC BY 2.0. https://www.flickr.com/photos/opalsson/18963844186/

OUR MISSION

Our ambition is to **collect**, **preserve**, and **share all software** that is publicly available in source code form. On this foundation, a wealth of applications can be built, ranging from cultural heritage and education to industry, from science to public administration, and more.

> *"Programs are written for people to read, and only accessorily for machines to execute"*
>
> — Harold Abelson

Collect

Software is the fabric that binds together our digital lives. Any software component may turn out to be essential in the future, so we **collect all software** that is publicly available in source code form, and we will encourage the construction of **curated archives** on top of Software Heritage.

We keep track of the **origin of software** we archive and store its full development history: this precious meta-information will be carefully harvested and structured for future use.

Preserve

Software is fragile and we are unfortunately starting to lose it, sometimes massively, when popular code hosting platforms shut down or reduce operations.

We preserve software, because **it contains** our technical and scientific knowledge. We preserve software because **it is the means of accessing** all of our knowledge. We know that for this to be sustainable, a **vast collective effort** is needed, and we will release as **free/open source software** all the software we write for the needs of Software Heritage and openly describe our technical architecture and processes.

We **are building** an open **network of peers** and mirrors that share with us the responsibility of maintaining several copies of all the software we collect.

Share

We are building the largest archive of software source code ever assembled. We will **index**, **organize**, **make referenceable and accessible** all of this precious heritage.

We provide **the SWHID unique identifiers**, intrinsically bound to the software components, and that need **no central registry**, to ensure that a resilient web of knowledge can be built on top of the Software Heritage archive.

A variety of services, ranging from documentation to classification, from search to distribution, will progressively be developed to release all the potential of this **Library of Alexandria of Software**.

SOFTWARE HERITAGE IN A NUTSHELL

We are building an essential infrastructure, that is meant to ensure three main properties for the source code we collect:

Availability

The code will be stored, preserved and made accessible on the long term.

Traceability

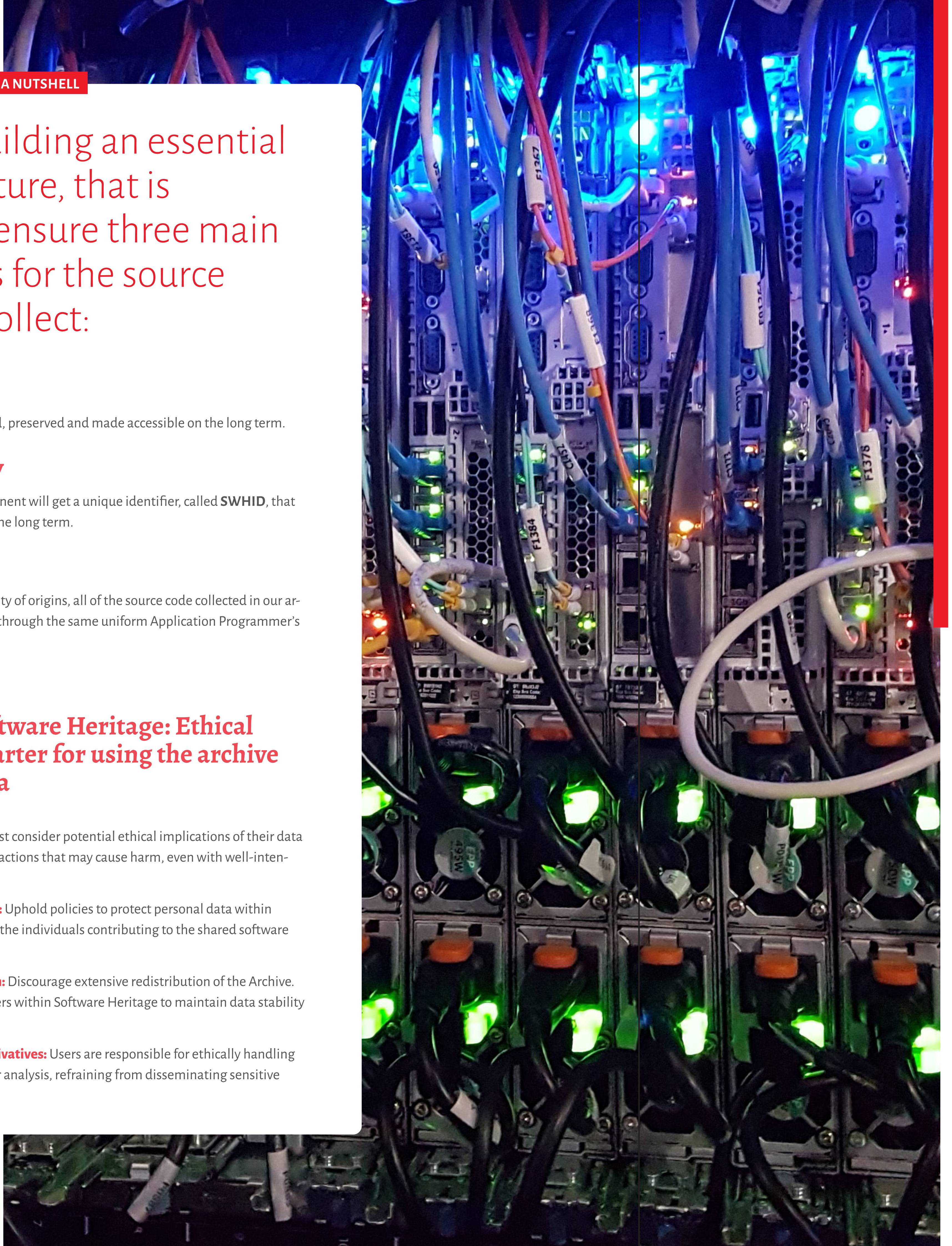
Eeach software component will get a unique identifier, called **SWHID**, that can be relied upon in the long term.

Uniformity

Despite the great variety of origins, all of the source code collected in our archive will be accessed through the same uniform Application Programmer's Interface (**API**)

Software Heritage: Ethical Charter for using the archive data

1. **Avoid Harm:** Users must consider potential ethical implications of their data usage, refraining from actions that may cause harm, even with well-intentioned research.
2. **Protect Personal Data:** Uphold policies to protect personal data within the archive, respecting the individuals contributing to the shared software commons.
3. **Minimize Distribution:** Discourage extensive redistribution of the Archive. Use persistent identifiers within Software Heritage to maintain data stability over time.
4. **Responsible Data Derivatives:** Users are responsible for ethically handling derived data from their analysis, refraining from disseminating sensitive information.



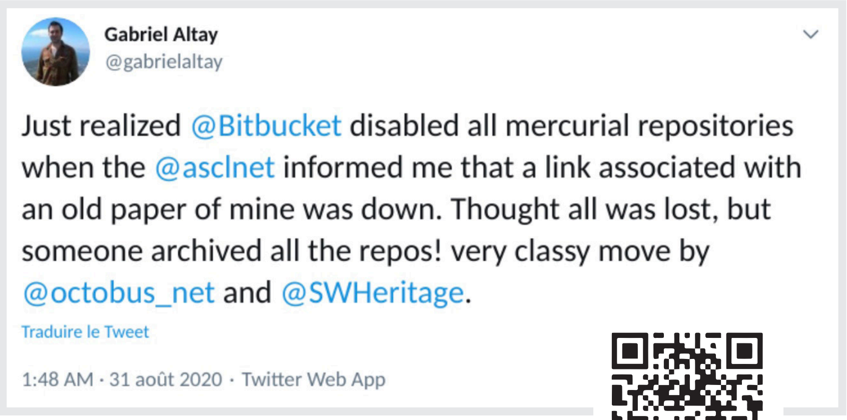
A catalog to find them all

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.

Adullact
Sourceforge
GoogleCode
CTAN
GitLab
Debian
GitHub
Maven
CRAN
CPAN
Inria
BerliOS
Bitbucket
Gitorious

An archive to preserve them

Modern software development relies on collaborative platforms, and many of them can be used free of charge. One can **create**, but also **modify** or **delete** projects: *they are not archives*. In recent years, we have seen several platforms come and go, sometimes suddenly, endangering hundreds of thousands of software projects all at once. Software Heritage is building the **universal archive** that is needed to ensure we will not loose source code any more.



Gabriel Altay
@gabrielaltay

Just realized @Bitbucket disabled all mercurial repositories when the @asclnet informed me that a link associated with an old paper of mine was down. Thought all was lost, but someone archived all the repos! very classy move by @octobus_net and @SWHeritage.

Traduire le Tweet

1:48 AM · 31 août 2020 · Twitter Web App

An instrument to explore and study them

Software underlies all aspects of our modern societies, and in a few decades we have built software systems of incredible complexity: some are huge programs, with tens of millions of lines of code, some are smaller programs, but mostl rely on hundreds or thousands of other components. We need to master this complexity, in order to build better, safer systems, and protect against malware.

Humankind has been able to build marvelous instruments to explore the universe, now it's time to build a common, shared infrastructure to explore and study the galaxy of software development. With enough support, Software Heritage can evolve into such an infrastructure.

HUMANKIND HAS BEEN ABLE TO BUILD MARVELOUS INSTRUMENTS TO EXPLORE THE UNIVERSE, NOW IT'S TIME TO BUILD A COMMON, SHARED INFRASTRUCTURE TO EXPLORE AND STUDY THE GALAXY OF SOFTWARE DEVELOPMENT.

TECHNOLOGY HIGHLIGHTS

All the public code history in a giant graph

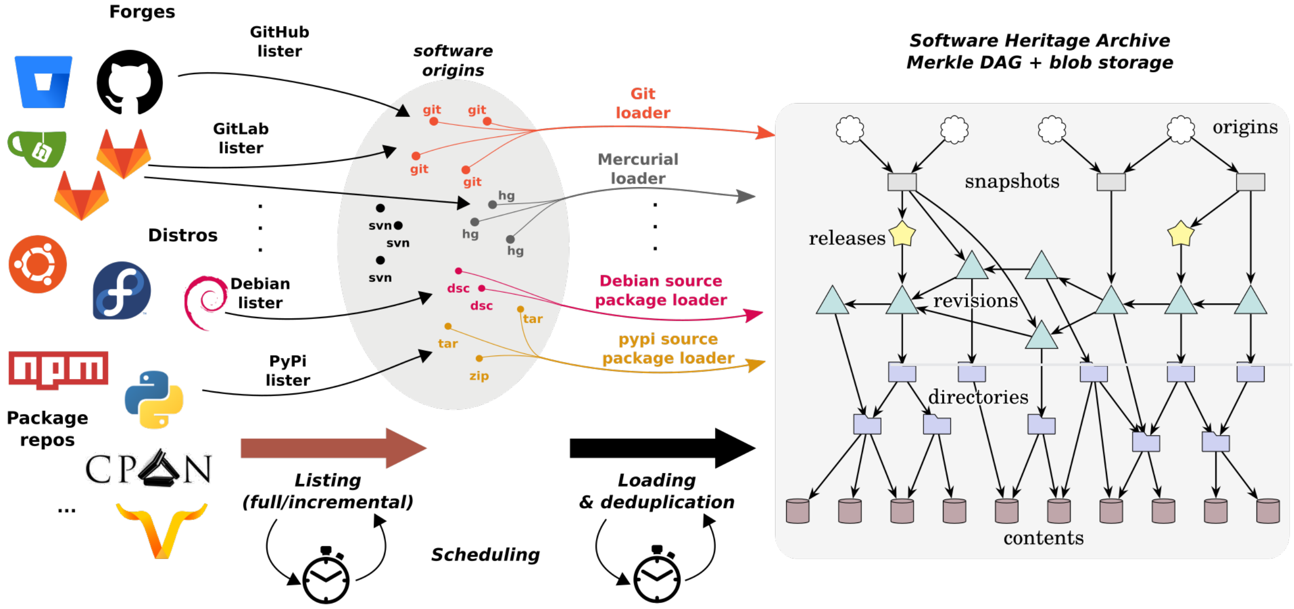
Merkle graphs and SWHID

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.

35 billion nodes

500 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 is separated into three phases: ***listing software sources***, ***scheduling updates*** and ***collecting the software artifacts*** into the archive.

Software Heritage Loaders

A loader is a software component used to ingest a software artifact into the *Software Heritage* archive, performing the appropriate conversion into the Merkle graph.

In 2022 we have unveiled a dedicate page with all available loaders and links to their high-level documentation: https://docs.softwareheritage.org/user/loaders.html

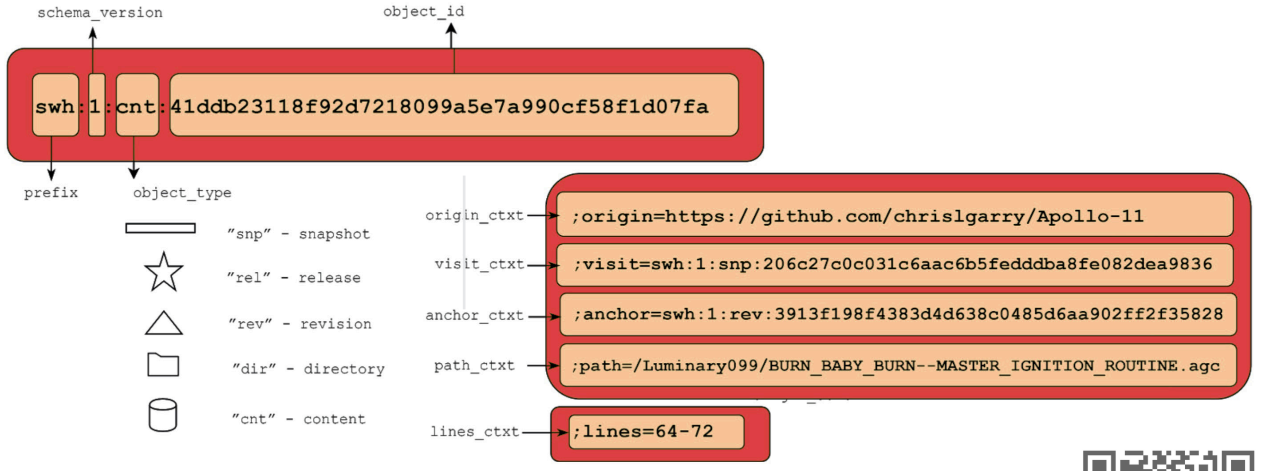
Software Heritage Listers

A lister is a software component used for discovering all software projects available on a code hosting or distribution platform. In 2022 we have unveiled a dedicate page with all the available listers and links to their high-level documentation: https://docs.softwareheritage.org/user/listers.html

The SWHID intrinsic persistent identifiers

All artefacts in the Software Heritage archive get a **SoftWare Hash IDentifier**, or **SWHID** for short, that is guaranteed to remain stable 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 everybody.


schema_version
object_id
swh:1:cnt:41ddb23118f92d7218099a5e7a990cf58f1d07fa
prefix
object_type
"snp" - snapshot
"rel" - release
"rev" - revision
"dir" - directory
"cnt" - content
origin_ctxt
visit_ctxt
anchor_ctxt
path_ctxt
lines_ctxt
;origin=https://github.com/chrislgarry/Apollo-11
;visit=swh:1:snp:206c27c0c031c6aac6b5fedddba8fe082dea9836
;anchor=swh:1:rev:3913f198f4383d4d638c0485d6aa902ff2f35828
;path=/Luminary099/BURN_BABY_BURN--MASTER_IGNITION_ROUTINE.agc
;lines=64-72

https://www.softwareheritage.org/2020/07/09/intrinsic-vs-extrinsic-identifiers/ →

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 the different entities, that are designated using systems of identifiers that come 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 ago 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 at all on the version control system: any software artifact ingested in the Software Heritage archive gets these identifiers.

SWHIDs are now part of the SPDX 2.2 industry specification, and have corresponding properties in Wikidata. A normalization process is underway.

WHAT MAKES SWHIDS SPECIAL IS THAT THEY DO NOT DEPEND AT ALL ON THE VERSION CONTROL SYSTEM

Image's credit ktsimage Stock photo ID:658008000

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.

The working group has release the SWHID Specification Version 1.1 in November 2023.

The first SWHID publicly available specification is out!

The precise identification of software artifacts and versions holds immense significance across various sectors, driving Software Heritage's core mission to collect, preserve, and share software source code. Utilizing the SWHID (Software Hash IDentifier) for over 30 billion software artifacts, Software Heritage ensures unambiguous referencing and retrieval, facilitating preservation efforts. Through an open process, the SWHID Working Group has crafted a comprehensive specification, recently approved and available online. The evolved term "Software Hash IDentifier" emphasizes its relevance beyond source code and Software Heritage. This milestone marks the beginning, not the end, inviting stakeholders to contribute and shape SWHIDs' evolution by proposing features that align with their use cases, fostering a collaborative, adaptable identification framework.

SWHID WORKING GROUP
swhid.org →

SWHID KICKOFF PRESENTATION
https://hal.science/hal-04121507 →

Services that Software Heritage offers today

Browse & Search

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

https://archive.software-heritage.org/

SWHID provider & resolver

SWH provides 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.

Go to the resolver API endpoint

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 directory API endpoint

https://archive.software-heritage.org/browse/vault/

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.

Go to API endpoint

https://save.softwareheritage.org/

Deposit

The deposit feature is a SWORD 2.0 Server implementation. **S.W.O.R.D** (**S**imple **W**eb-Service **O**ffering **R**epository **D**eposit) is an interoperability standard for digital file deposit. The deposit allows a client (a repository, e.g. HAL) to submit software source archives and its associated metadata to the SWH archive. Metadata can be also submitted referencing a repository url (origin) or a SWHID.

For more information

https://deposit.softwareheritage.org

Add Forge Now

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

https://docs.software-heritage.org/devel/swh-lister/tutorial.html#lister-tutorial

NEW IN 2023

GraphQL

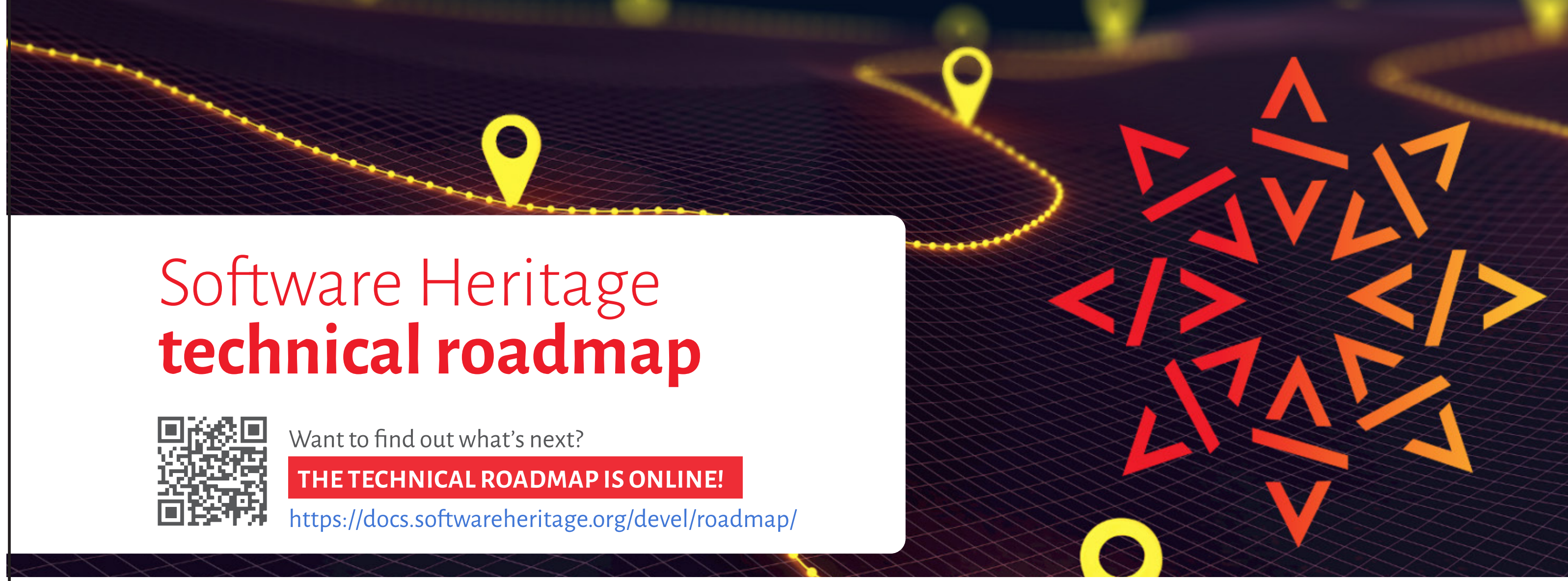
GraphQL allows a client to fetch the server data using a query language and enables them to create powerful requests.

https://www.softwareheritage.org/2023/07/25/software-heritage-graphql-explorer-more-power-to-your-apis/

Webhooks

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

https://www.softwareheritage.org/2023/06/01/webhooks-integrate-swh-workflow/



Software Heritage **technical roadmap**

Want to find out what's next?

THE TECHNICAL ROADMAP IS ONLINE!

https://docs.softwareheritage.org/devel/roadmap/

Image's credit Christoph Burgstedt Stock photo ID:1264076971

BROWSER EXTENSION

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

READ THE DOCS

https://docs.softwareheritage.org/

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.

https://archive.software-heritage.org/api/

Architecture

Archiving a repository from a forge isn't the same action as archiving source code from a package manager. It becomes even harder when you realize that version control systems have evolved a lot over the last decades. The SWH architecture was designed to harmonize different sources into a robust infrastructure.

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. Using also origins, visits ans snapshots to store provenance information.

Read more in Software Heritage: Why and How to Preserve Software Source Code.

Metadata

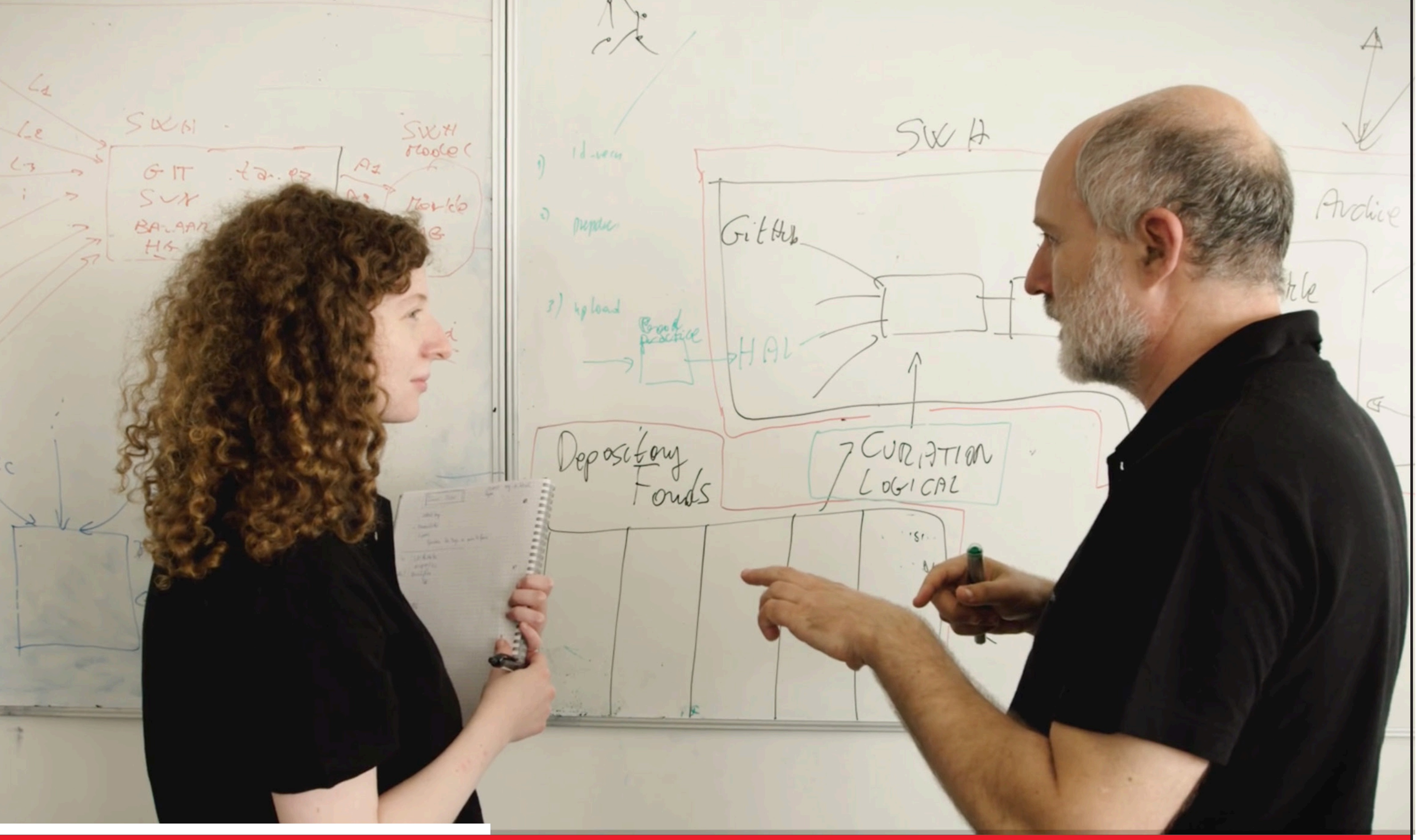
SWH collects and extracts metadata that describes and provides additional information on source code.

- Extrinsic metadata are metadata which aren't 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

swh-indexer module is in charge for computing source code files to extract information with the following objectives:

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



A shared infrastructure for multiple stakeholders

Culture and education

> *"[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**.

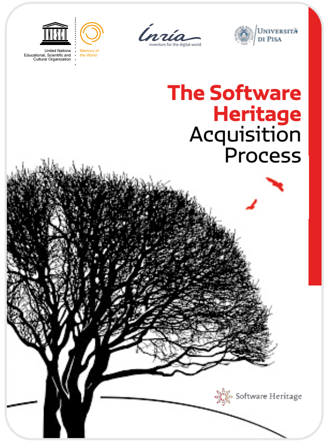
It is an important part of our heritage that we cannot afford to lose. Software is furthermore a key enabler for preserving other parts of our cultural heritage that we would de facto lose if we lose the software needed to access them. Preserving software is essential for preserving our cultural heritage.

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.

"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."

Let's Not Dumb Down the History of Computer Science
Donald E. Knuth, Len Shustek
https://doi.org/10.1145/3442377

[1] Available from the UNESCO website as ark:/48223/pf0000366715, 2019.



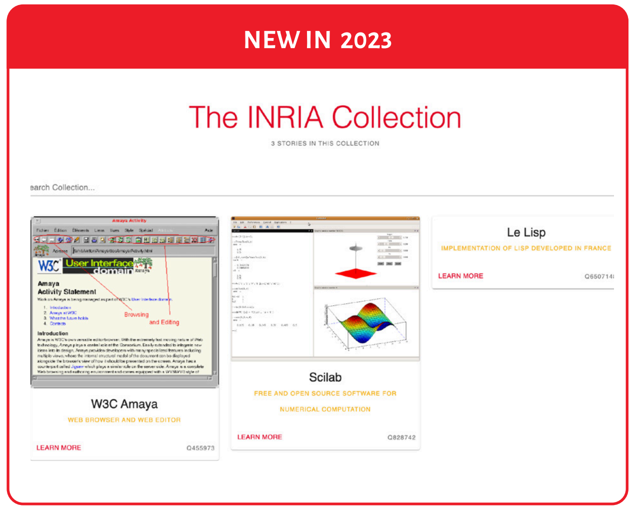
The Software Heritage Acquisition Process

Software Heritage

Software Heritage Acquisition Process

Rescuing landmark legacy source code

The Software Heritage Acquisition Process (**SWHAP**), developed in collaboration with UNESCO and the University of Pisa, details all the steps needed to successfully curate landmark legacy source code and archive it in Software Heritage.



NEW IN 2023

The INRIA Collection

3 STORIES IN THIS COLLECTION

earch Collection...

W3C Amaya
WEB BROWSER AND WEB EDITOR
LEARN MORE Q455973

Scilab
FREE AND OPEN SOURCE SOFTWARE FOR NUMERICAL COMPUTATION
LEARN MORE Q828742

Le Lisp
IMPLEMENTATION OF LISP DEVELOPED IN FRANCE
LEARN MORE Q650714

Software Stories

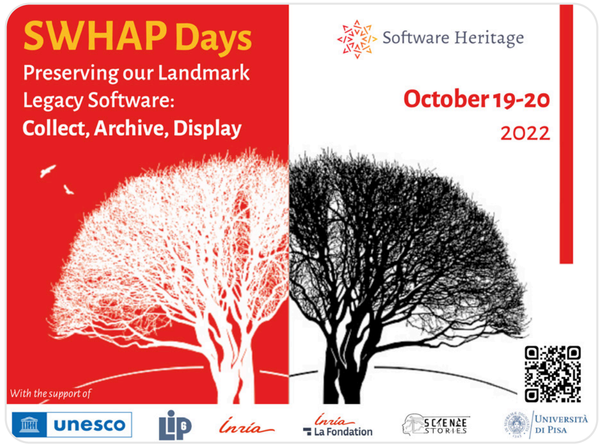
Highlighting the human side behind the software projects

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.

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 aims to generate a comprehensive overview of the



SWHAP Days & Workshop

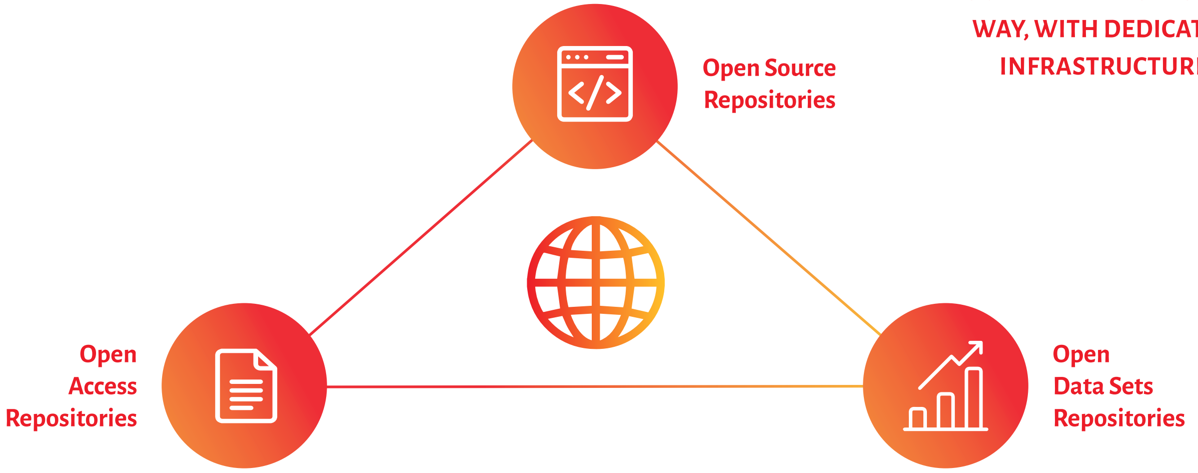
In October 2022, Software Heritage hosted its inaugural SWHAP Days, a two-day conference dedicated to software preservation. In 2023, the Software Heritage team decided to organize a two-day, hands-on workshop in a closed committee format to further delve into practical aspects and foster collaborative discussions on advancing software preservation efforts.

SWHAP workshop, 2023 Proceedings →

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.

“SOFTWARE SOURCE CODE IS MUCH MORE THAN DATA, IT IS A CREATION OF THE HUMAN INGENUITY, AND RESEARCH SOFTWARE NEEDS TO BE ARCHIVED, REFERENCED, DESCRIBED AND CREDITED IN A SPECIFIC WAY, WITH DEDICATED INFRASTRUCTURES”.



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

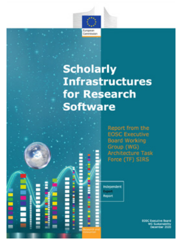
“Open Scientific knowledge [includes] open source software: source code must be included in the software release and made available on openly accessible repositories”.

– Unesco recommendation for open science,
November 2021

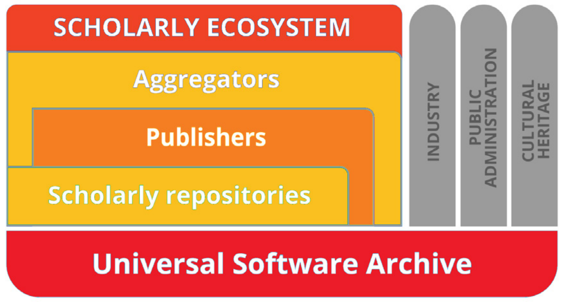
Serving the scholarly ecosystem

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 needs to be archived, referenced, described and credited in a specific way, with dedicated infrastructures.



A multi-year collaboration between the french national open access portal HAL and Software Heritage has led to developing a seamless workflow to archive, reference, describe and cite research software, and the Second National Plan for Open Science now recommends that all french researchers use it *and fixes the objective to standardize* the SWHID identifiers. *In February 2022, Software Heritage has been inscribed in France in the national roadmap of research infrastructures.*

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.

IPOL & eLife

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

European Projects

Our goal, through participation in EU 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.

Within the FAIRCORE4EOSC project, the Beta version of the Research Software APIs and connectors was released in November 2023, seamlessly interconnecting research output infrastructures with the Software Heritage universal source code archive.

FAIR-IMPACT European project, launched in June 2022, has the role to support and disseminate 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.

Adoption and Recognition

The real antidote [to epidemic] is scientific knowledge and global cooperation.
– Yuval Noah Harari (on COVID 19)

Joining the SCOSS family

The Global Sustainability Coalition for Open Science Services (SCOSS) board has selected Software Heritage for its 5th pledging round, in November 2023, recognizing Software Heritage as a crucial open science infrastructure ensuring continuous access to the software code outputs generated by researchers worldwide.
SCOSS Members, libraries, archives, institutions and research funders supporting open science can make a difference by committing to fund Software Heritage.
Pledge an annual donation for three years, offering a secure financial foundation and access to the dedicated Software Heritage Archives and Libraries Interest Group (ALIG).
Discover the detailed pledging program in the section dedicated to the 5th pledging round on the official SCOSS website.

Running software, again and again

Software Heritage ensures the availability and traceability of software source code, a key prerequisite for reproducing, reusing and adapting existing software. Partnership are being established to connect Software Heritage with package managers and build systems, to enable the replication of full blown executables and systems, the ultimate goal of reproducibility.
Guix is an advanced distribution of the GNU operating system developed by the GNU Project that has made reproducibility its core mission, setting it apart from other tools. It is the first free software distribution backed by a stable archive (a detailed presentation of what goes on under the hood can be found on the Guix blog.

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.

BibLaTeX

The Biblatex-software package lets you produce beautiful bibliographic entries for software, and it supports SWHID natively. Biblatex-software is integrated in CTAN and TeXLive, and works out of the box in Overleaf. As of April 2022, biblatex-software is integrated in the ACM article style.

French research strategy

Software Heritage has been selected to be included in the french national strategy for research infrastructure, a recognition of the key role that it plays, together with the HAL open access portal, for archiving, referencing, describing and citing research software.
The french national research funding agency recommends Software Heritage for all funded projects.

Funding agencies recommendations
ANR 2023 guidelines (p. 17)

MIRRORS

Towards a global infrastructure for ***research on software source code***

"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

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, but we also look at **decentralized** technologies.

In order to prevent information loss, and simplify access to humankind's software heritage, we are building an international network of ***mirrors***.

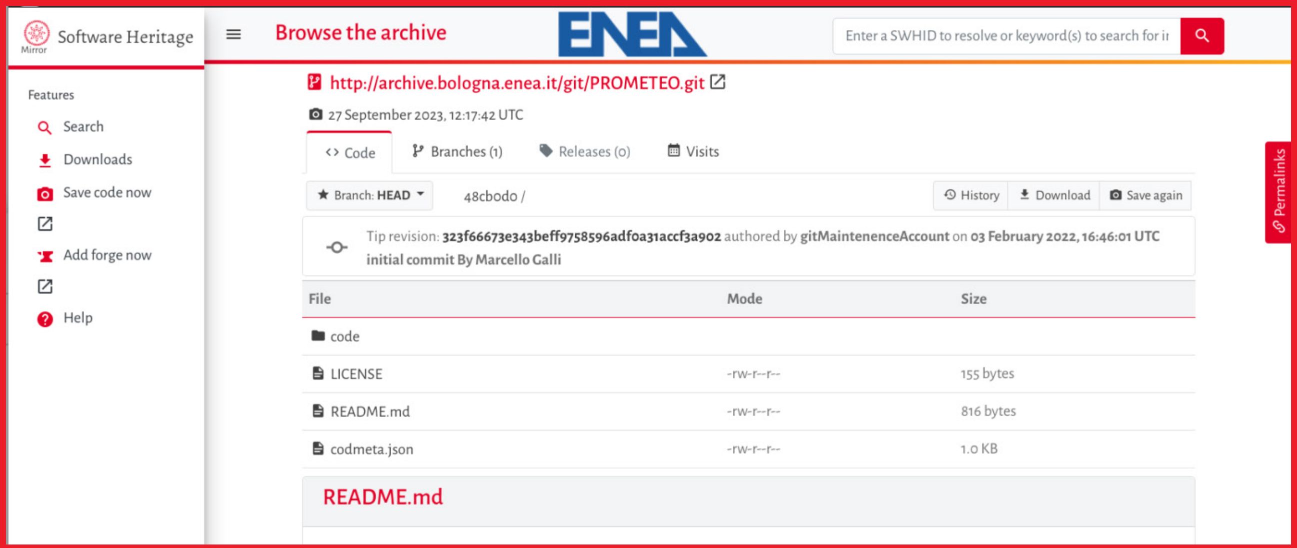
A ***mirror*** is a full copy of the Software Heritage universal source code archive, operated ***in agreement with***, but ***independently from*** the Software Heritage organization.

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.

ENEA opens the first Software Heritage Mirror

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



http://archive.bologna.enea.it/git/PROMETEO.git
27 September 2023, 12:17:42 UTC

48cbodo /

Tip revision: **323f66673e343beff9758596adf0a31accf3a902** authored by **gitMaintenenceAccount** on **03 February 2022, 16:46:01 UTC**
initial commit By Marcello Galli

File	Mode	Size
code		
LICENSE	-rw-r--r--	155 bytes
README.md	-rw-r--r--	816 bytes
codmeta.json	-rw-r--r--	1.0 KB

README.md

Mirror ethical charter →

Research on the Largest Public Archive of Software Source Code

The Software Heritage Graph Dataset

The Software Heritage Graph Dataset is a fully deduplicated Merkle DAG representation of the Software Heritage Archive. The dataset links together file contents identifiers, source code diretories, Version Control System (VCS) commits tracking evolution over time, up to the full states of VCS repositories ass observed by Software Heritage during perioddic crawls. The Dataset's contents coome from major development forges (including GitHub and GitLab), FOSS distributinos (e.g., Debian), and language-specific package managers (e.g., PyPI).

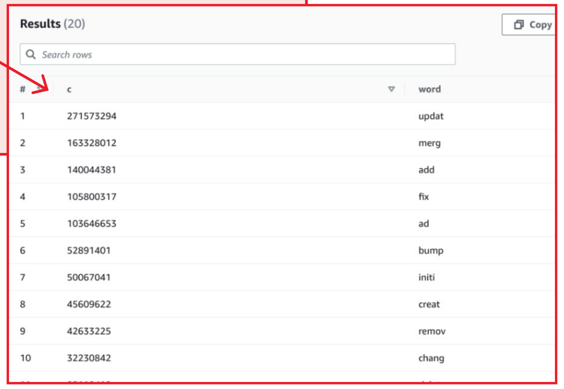
We publish a relational representation of the full archive of Software Heritage as a set of tables. Available as open data in the AWS Open Dataset collection, it makes it easier for researchers to perform large-scale reproducible software studies.

Here is a sample query to find the most popular commit verbs across all the archive.

QUERY

```
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;
```

RESULTS



Results (20)

#	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: Proceedings of the 16th International Conference on Mining Software Repositories, pp. 138-142, IEEE Press, 2019.

Loading the Software Heritage graph in memory

With ***over 25 billion nodes and over 350 billion edges***, the Software Heritage graph is one of the largest public social graphs available. Thanks to bleeding edge graph compression technology, it can now all fit in 200Gb of memory, and be traversed in ***just tens of nanoseconds per edge***!
Java and gRPC APIs available:
https://docs.softwareheritage.org/devel/swh-graph/grpc-api.html

Paolo Boldi, Antoine Pietri, Sebastiano Vigna, Stefano Zacchiroli
Ultra-Large-Scale Repository Analysis via Graph Compression
SANER 2020, 27th Intl. Conf. on Software Analysis, Evolution and Reengineering. IEEE

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.

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

Key findings

Large reproducible datasets

The lack of reproducibility is a significant program in computer science. We can leverage the Software Heritage archive to build very large, fully reproducible datasets for software engineering research.

Lefeuvre, Galasso, Combemale, Sahraoui, Zacchiroli
Fingerprinting and Building Large Reproducible Datasets
ACM-REP 2023

Diversity, equality, 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, but the COVID-19 pandemic has reversed the trend.

Zacchiroli. **Gender differences in public code contributions: a 50-year perspective.** IEEE Software, 2021

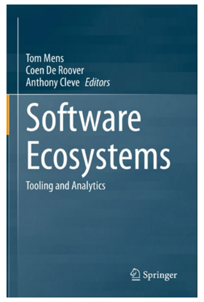
Rossi and Zacchiroli. **Worldwide gender differences in public code contributions.** ICSE SEIS, 2022

Rossi and Zacchiroli. **Geographic diversity in public code contributions.** MSR 2022

Detecting project *forks*

Today, developers contribute to open-source projects by working on their own copies, called ***forks***, that can be created in many ways. The Software Heritage archive allows to detect "exogenous" forks across multiple platforms.

Pietri, Rousseau, Zacchiroli.
Forking Without Clicking: on How to Identify Software Repository Forks. MSR 2020



The Software Heritage Open Science Ecosystem

Learn about it in a dedicated chapter in the 2023 book about Software Ecosystems research

R&D Prototyping

The bridge built by Obsidian, opens the way to a solution for IPFS users to access, retrieve, and redistribute SWH archival resources, while leveraging the P2P network to ease the burden of distribution for Software Heritage.

Industry

[There is a need to] *ensuring and attesting, to the extent practicable, to the* ***integrity and provenance*** *of open source software used within any portion of a product.*

Executive Order on Improving the Nation's Cybersecurity
White House, May 12, 2021

The ability to use, understand and evolve the processes and devices that run our industry relies on the ability to access, understand, and evolve the software that controls them.
Software Heritage provides a *neutral, common, shared, open, non-profit*, **reference knowledge base** encompassing all the software source that is publicly available, enabling new applications to improve all aspects of the software process.

"THESE UNIQUE FEATURES OF SOFTWARE HERITAGE ENSURE AVAILABILITY, GUARANTEE INTEGRITY AND ENABLE TRACEABILITY OF THE SOURCE CODE OF ALL ARTEFACTS IN THE OPEN SOURCE SOFTWARE SUPPLY CHAIN".

Securing the Open Source software supply chain

The *uniform, technology-neutral global Merkle graph* provides, with the growing mirror network, **a transparent source of trust.** The SWHID provides *uniform, technology-independent, and cryptographically strong* **intrinsic identifiers** to *track source code artifacts at all levels*.
These unique features of Software Heritage ensure **availability**, guarantee **integrity** and enable **traceability** of the source code of all artefacts in the open source software supply chain.

Complete and corresponding source code

Industry players can **delegate** to the Software Heritage archive the task to preserve and make available to third parties the complete and corresponding source code of any open source component they use. They can use the Software Heritage as a trusted reference in their agreements.

Enhancing Cybersecurity through Software Heritage

Software Heritage participated in the launch of a new national research and innovation program on cybersecurity - PTCC. During the opening event, Roberto Di Cosmo, Software Heritage's director, unveiled the first project funded in this framework, known as SWHSec. This groundbreaking initiative brings together eight expert research teams specializing in security, software engineering, and open-source software to harness the power of Software Heritage's robust infrastructure and create cutting-edge tools for cybersecurity.

Compliance for source code distribution

Software Heritage simplifies source code distribution

Companies adopting Software Heritage can effectively *outsource their source code distribution obligations.* Given its mission of long term software preservation, Software Heritage is the **perfect steward** for the source code it receives.

Software Heritage provides perpetual identifiers

Software Heritage provides uniform, cryptographic identifiers for all of the 10 billion software artefacts it archives. These *intrinsic* identifiers allow to *independently* verify the integrity of the software artefacts they denote.
Without having to trust any third party.

Software Heritage can be integrated in your compliance process, saving time and resources

Source code deposit in Software Heritage can be integrated in any compliance process. As soon as code is deposited, you get the corresponding intrinsic identifier: that's the only piece of information you will ever need to be compliant. Software Heritage **takes over** all other distribution obligations from there. By contrast, typical in-house industry approaches require to maintain dedicated infrastructure and resources
over a very long term.

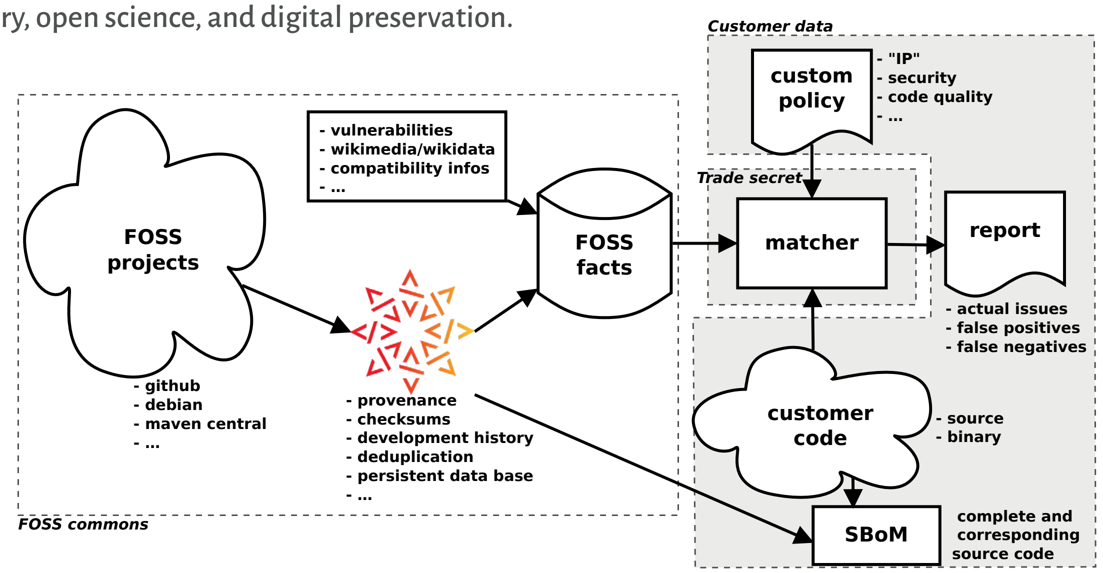
Software Heritage allows you to rely on a mutualised infrastructure

When you join Software Heritage, you adopt an **open infrastructure** that is backed by a growing wordlwide community that brings together stakeholders in **industry, open science, and digital preservation.**

How it works

1. You prepare the complete and corresponding source code (CCS) archive, as usual.
2. You deposit the archive into the Software Heritage platform, using an API that you can integrate with your continuous delivery process.
3. You get back a perpetual identifier that anybody can use to retrieve and browse the source code you deposited.
4. You are happy to know that SoftWware Heritage preserves the archive and will host it forever.

And yes, this seamless process, is fully compliant with all major copyleft licenses!



Contact us about your compliance needs at compliance@softwareheritage.org

Empowering Public Administration

Promoting the sharing of open source solutions created or used by administrations within the European Union [...] results in enhanced collaboration between public administrations

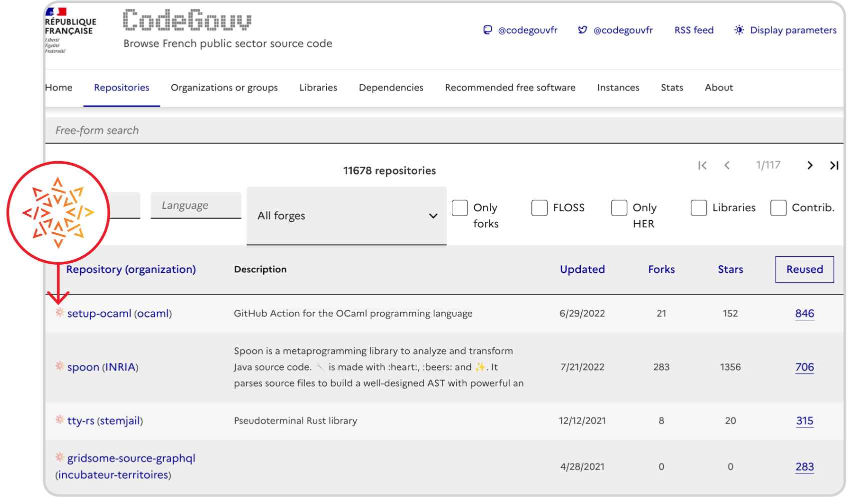
Strasbourg Declaration,
May 2022, European Union

Fostering Transparency and Efficiency in the Digital Age

Public administrations strive to make their action transparent to the citizens, while improving the services they provide by sharing and reusing their software.

Transparency and Long-Term Availability

Software Heritage provides the one-stop archive where all public software can be deposited and referenced, open to all, with the guarantee that it will not disappear.
The Open Source mission in the French DINUM uses Software Heritage to systematically archive the open source software of the french public administrations.



RÉPUBLIQUE FRANÇAISE
Liberté Égalité Fraternité

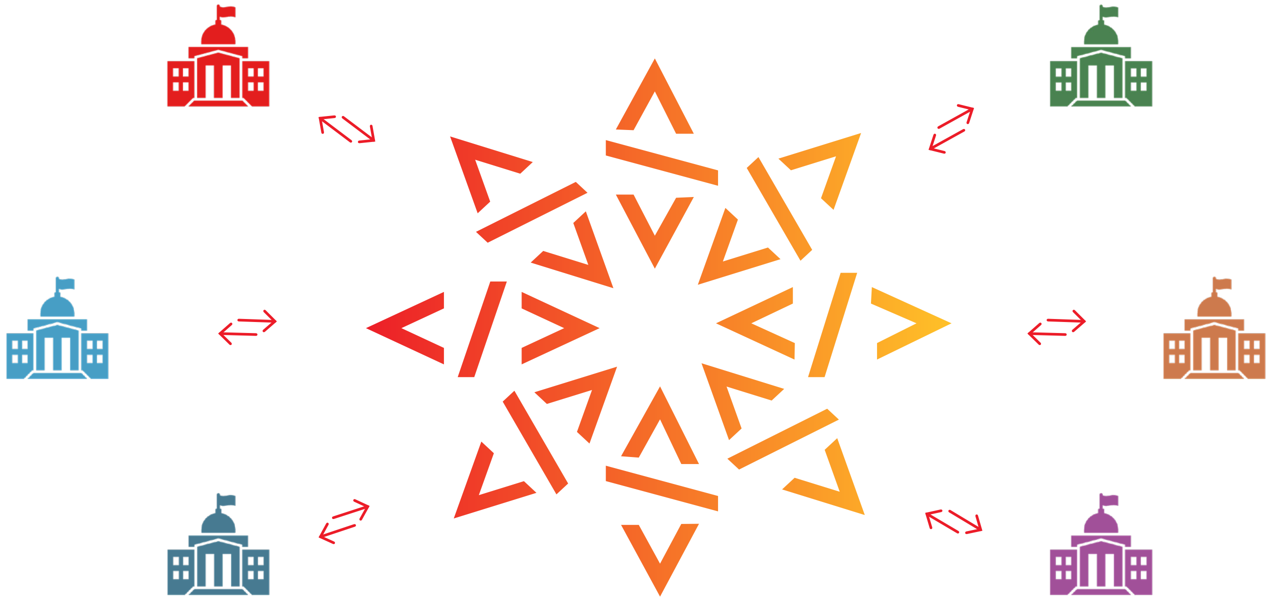
CodeGouv
Browse French public sector source code

11678 repositories

Repository (organization)	Description	Updated	Forks	Stars	Reused
setup-ocaml (ocaml)	GitHub Action for the OCaml programming language	6/29/2022	21	152	846
spoon (INRIA)	Spoon is a metaprogramming library to analyze and transform Java source code. 🥄 is made with :heart:, :beers: and ✨. It parses source files to build a well-designed AST with powerful an	7/21/2022	283	1356	706
tty-rs (stemjail)	Pseudoterminal Rust library	12/12/2021	8	20	315
gridsome-source-graphql (incubateur-territoires)		4/28/2021	0	0	283

Deposit and Sharing of Metadata

Public administrations can deposit qualified metadata in the Archive, in machine readable form, enabling sharing and reuse of information across administrations, countries and continents.



Software Heritage Statement on Large Language Models for Code

As we strive to preserve this vital resource for future generations, we acknowledge the emergence of inquiries regarding the use of the Software Heritage archive for the training of machine learning models, particularly large language models (LLMs) that can automatically generate code to assist with software development tasks.

In alignment with our mission, we believe that LLMs for code should be built in a **transparent and respectful way, to the benefit of all**. We hence state the following principles for acceptable machine learning use of the Software Heritage archive.

Principles

1. 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**, together with the documentation and toolings needed to use them.
2. The **initial training data** extracted from the Software Heritage archive must be **fully and precisely identified** by, for example, publishing the corresponding SWHID identifiers (note that, in the context of Software Heritage, public availability of the initial training data is a given: anyone can obtain it from the archive). This will enable use cases such as: studying biases (fairness), verifying if a code of interest was present in the training data (transparency), and providing appropriate attribution when generated code bears resemblance to training data (credit), among others.
3. Mechanisms should be established, where possible, **for authors to exclude** their archived code from the training inputs before model training begins.

Building for the long term

Building a global infrastructure to stand the test of time is a humbling undertaking. To this end, we rely on the following founding principles

Experience shows that a single for profit entity, however powerful, does not provide sufficient durability guarantees in the long term. We believe that it is essential to build a **non profit multi-stakeholder** foundation that has the mission of Software Heritage as its explicit primary objective, and we are delighted to be working with UNESCO towards it.

Transparency of code and architecture

Long-term preservation efforts cannot be based on black boxes that hide the process behind closed doors. We are long-time Free/Open Source Software developers and advocates, and our code and specifications are released under a **Free and Open Source Software** license.
We are designing a complex software architecture. Its design and specifications are public.

Collaborative development

The mission of Software Heritage is a humbling undertaking: to succeed, a large collective effort is needed. To foster it, we adopt an **open development** process, and strive to create an active community around all components of the Software Heritage infrastructure.

Facts and provenance

Following best archival practices, Software Heritage will store full provenance information, in order to be able to always state **what** was found **where** and **when**.

You can help

The Software Heritage archive will serve the needs of the many, from cultural institutions to scientists and industries. Everyone can help us achieving these ambitious goals and there are several ways to help.

Collaboration and community

Alone we go faster, together we go further.
African saying

A broad community is key for succeeding in the long-term mission undertaken by Software Heritage. This is why we are partnering with private funders around the world to provide grants for experts that are willing to engage with the long-term mission of Software Heritage.

Alfred P. Sloan Foundation

A grant from Alfred P. Sloan Foundation has been awarded to Software Heritage specifically to foster the emergence of a community of expert contributors to increase the coverage of the Software Heritage archive. **Seven subgrants** have been distributed, resulting in over 300.000 new repositories being archived.

NGI Zero

Four cascading grants from the NLNet Foundation funded work that allowed Software Heritage to save 250.000 endangered Bitbucket repositories, improve its Mercurial loader, get connectors with Nix and Guix, and experiment with the IPFS distributed file system.

NGI Search

The NGI Search project is a European project designed to support entrepreneurs, tech-geeks, developers, and socially engaged people, who are capable of challenging the way we search and discover information and resources on the internet.

Become a sponsor

Pursuing our roadmap for the archive requires significant resources. We welcome companies, institutions, and individuals who would like to join our sponsorship program and sustain the Software Heritage project.

Tackle scientific challenges

Building, maintaining, and exploiting the universal source code archive poses relevant scientific challenges. We welcome scientists who would like to contribute to this mission by participating in our research activities.

Code with us

All the software we develop ourselves is open source. We welcome contributors that are willing to delve into it and help us building the many components that are needed to make the archive progress towards the next milestones.

Users

Find all user-related tools and features to guide you in your Software Heritage journey. Connect, share, and engage with the community to enrich and help us build the universal source code archive.

Grantees

Castalia Solutions Elegant Software Engineering · OCTOBUS · OCaml PRO · TWEAG by Modus Create

Meet our team

BEHIND SOFTWARE HERITAGE YOU FIND A TEAM OF PASSIONATE PEOPLE THAT DEDICATE ALL THEIR ENERGY TO THE LONG TERM MISSION OF COLLECTING, PRESERVING AND SHARING THE SOURCE CODE OF ALL PUBLICLY AVAILABLE SOFTWARE. HERE YOU FIND THE COMPOSITION OF THE TEAM AS OF DECEMBER 2023.

Executives

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

Advisors

Gérard Berry *(French Academy of Science)*
Jean-François Abramatic *(EIT)*
Julia Lawall *(Inria)*
Serge Abiteboul *(French Academy of Science)*

Management

Benoît Chauvet *(Project Manager)*
David Douard *(Dev Team Manager)*
Morane Gruenpeter *(Head of Open Science)*
Vincent Sellier *(Sysadmin Team Manager)*

Engineers

Lunar (Jérémy Bobbio)
Nicolas Dandrimont
Antoine R. Dumont
Antoine Lambert
Valentin Lorentz
Guillaume Samson
Jayesh Velayudhan

Open science community manager Sabrina Granger

Communication Marla da Silva

Visiting scientists Mathilde Fichen

Interns Tommaso Fontana

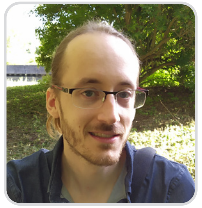
Visiting hackers

Kumar Shivendu
Stephan Sperling
Paul Wise

Ambassadors

At **Software Heritage**, We understand that success is achievable only through the collective efforts of a diverse community. Since 2020, the ambassador program has been instrumental in nurturing collaboration and promoting the widespread adoption of Software Heritage across various communities.
To foster community engagement, and accelerate the adoption of Software Heritage in the many fields where it brings groundbreaking benefits, a dedicated ambassador program has been established.

Agustín Benito Bethencourt


Alexis Lebis

Anna-Lena Lamprecht

Bertrand Néron

Borut Kumperscak

Bostjan Spetic

Camille Françoise

Bruno Khélifi

Cécile Arènes

Dare Pejić

Flavia Marzano

Frédéric Santos

Gavin Henry

Gerard Coen

Gilmary Gallon

Harish Pillay

Italo Vignoli

Jaime Arias

Joenio Marques Da Costa

Julien Caugant

Malin Sandström

Maria-Chiara Prodi

Max Kalik

Maxence Azzouz-Thuderoz

Mohammad Akhlaghi

Neal Fultz

Océane Valencia

Pierre Poulain

Sandrine Layrisse

Simon Phipps

Vicky Rampin

Violaine Louvet

Wendy Hagenmaier

Becoming an Ambassador

Interested in becoming a Software Heritage ambassador? Please tell us a bit about yourself and your interest in the mission of Software Heritage.

ambassadorprogram@softwareheritage.org

Software Heritage will provide solid, common foundations to serve the different needs of heritage preservation, science, and industry.

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