Software Heritage

key infrastructure for Open Science and Software Science

Jaime Arias

Research Engineer
CNRS, LIPN, Université Sorbonne Paris Nord

November 27, 2024

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

Who am I?

Hello!

I am **Jaime Arias**

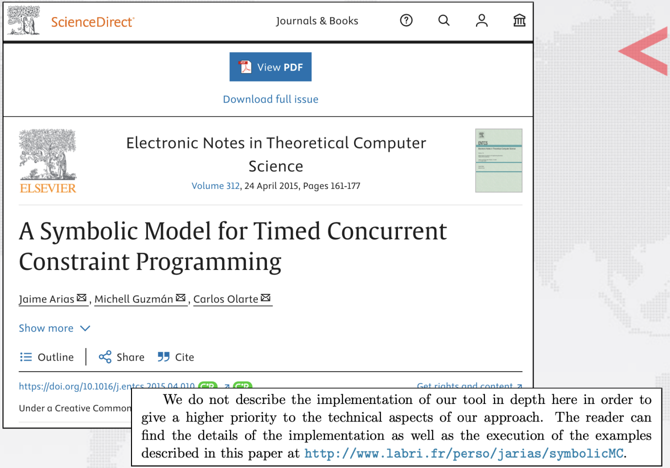
- CNRS Research Engineer @ LIPN
- Member @ Collège Codes Sources et Logiciels
- Ambassador @ Software Heritage

You can find me at:

✉ arias@lipn.fr

🔗 https://www.jaime-arias.fr

TL;DR



Electronic Notes in Theoretical Computer Science

Volume 312, 24 April 2015, Pages 161-177

A Symbolic Model for Timed Concurrent Constraint Programming

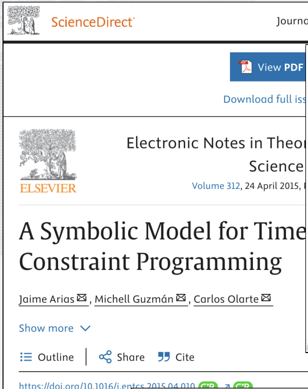
Jaime Arias, Michell Guzmán, Carlos Olarte

https://doi.org/10.1016/j.entcs.2015.04.010

Under a Creative Common

We do not describe the implementation of our tool in depth here in order to give a higher priority to the technical aspects of our approach. The reader can find the details of the implementation as well as the execution of the examples described in this paper at http://www.labri.fr/perso/jarias/symbolicMC.

TL;DR


ScienceDirect

Journals & Books

View PDF

Download full iss

Electronic Notes in Theo

Science

Volume 312, 24 April 2015,

A Symbolic Model for Time
Constraint Programming

Jaime Arias, Michell Guzmán, Carlos Olarte

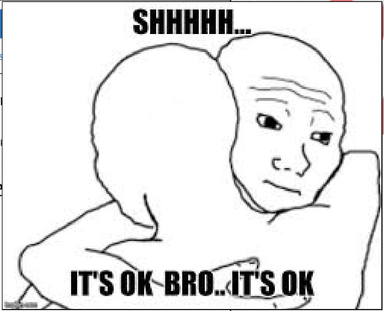
Show more

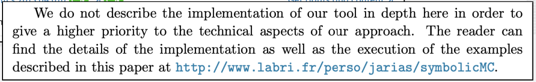
Outline | Share | Cite

https://doi.org/10.1016/j.entcs.2015.04.010

Get rights and content

Under a Creative Common



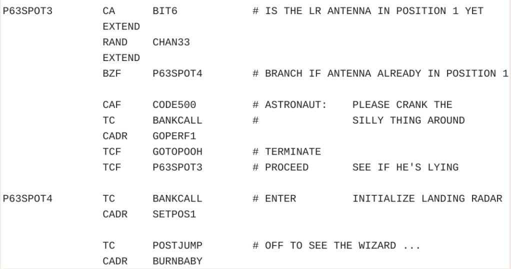

We do not describe the implementation of our tool in depth here in order to give a higher priority to the technical aspects of our approach. The reader can find the details of the implementation as well as the execution of the examples described in this paper at http://www.labri.fr/perso/jarias/symbolicMC.

Outline

1. Open Science & Software
2. Software Heritage for Open Science and Reproducibility
3. Quick Demo !
4. Call to action
5. Conclusion

Software *source code* is precious knowledge

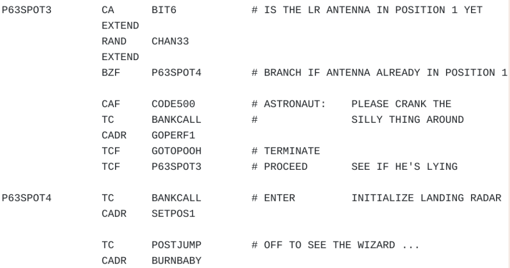
Apollo 11 source code (excerpt)



```
P63SPOT3        CA      BIT6            # IS THE LR ANTENNA IN POSITION 1 YET
                EXTEND
                RAND    CHAN33
                EXTEND
                BZF     P63SPOT4        # BRANCH IF ANTENNA ALREADY IN POSITION 1

                CAF     CODE500         # ASTRONAUT:     PLEASE CRANK THE
                TC      BANKCALL        #                SILLY THING AROUND
                CADR    GOPERF1
                TCF     GOTOPOOH        # TERMINATE
                TCF     P63SPOT3        # PROCEED        SEE IF HE'S LYING

P63SPOT4        TC      BANKCALL        # ENTER          INITIALIZE LANDING RADAR
                CADR    SETPOS1

                TC      POSTJUMP        # OFF TO SEE THE WIZARD ...
                CADR    BURNBABY
```

Software *source code* is precious knowledge

Apollo 11 source code (excerpt)



```
P63SPOT3        CA      BIT6            # IS THE LR ANTENNA IN POSITION 1 YET
                EXTEND
                RAND    CHAN33
                EXTEND
                BZF     P63SPOT4        # BRANCH IF ANTENNA ALREADY IN POSITION 1

                CAF     CODE500         # ASTRONAUT:     PLEASE CRANK THE
                TC      BANKCALL        #                SILLY THING AROUND
                CADR    GOPERF1
                TCF     GOTOPOOH        # TERMINATE
                TCF     P63SPOT3        # PROCEED        SEE IF HE'S LYING

P63SPOT4        TC      BANKCALL        # ENTER          INITIALIZE LANDING RADAR
                CADR    SETPOS1

                TC      POSTJUMP        # OFF TO SEE THE WIZARD ...
                CADR    BURNBABY
```

Quake III source code (excerpt)

```
float Q_rsqrt( float number )
{
    long i;
    float x2, y;
    const float threehalfs = 1.5F;

    x2 = number * 0.5F;
    y = number;
    i = * ( long * ) &y; // evil floating point bit level hacking
    i = 0x5f3759df - ( i >> 1 ); // what the fuck?
    y = * ( float * ) &i;
    y = y * ( threehalfs - ( x2 * y * y ) ); // 1st iteration
// y = y * ( threehalfs - ( x2 * y * y ) ); // 2nd iteration, this
can be removed

    return y;
}
```

Software *source code* is precious knowledge

Harold Abelson, Structure and Interpretation of Computer Programs (1st ed.) 1985

> *“Programs must be written for people to read, and only incidentally for machines to execute.”*

Software *source code* is precious knowledge

Harold Abelson, Structure and Interpretation of Computer Programs (1st ed.) 1985

"Programs must be written for people to read, and only incidentally for machines to execute."

Len Shustek, Computer History Museum 2006

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

Software *source code* is precious knowledge

Harold Abelson, Structure and Interpretation of Computer Programs (1st ed.) 1985

“Programs must be written for people to read, and only incidentally for machines to execute.”

Len Shustek, Computer History Museum 2006

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

Sonatype Survey 2017

80% to 90% of a new application is … just to **reuse**!

Software *source code* is precious knowledge

Harold Abelson, Structure and Interpretation of Computer Programs (1st ed.) 1985

"Programs must be written for people to read, and only incidentally for machines to execute."

Len Shustek, Computer History Museum 2006

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

Sonatype Survey 2017

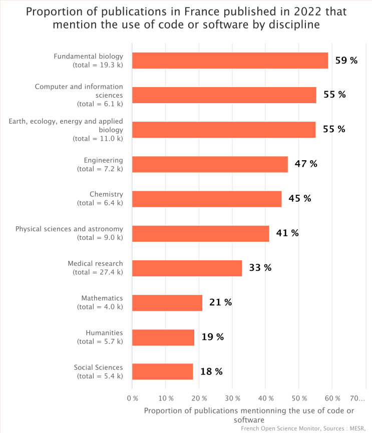
80% to 90% of a new application is … just to **reuse**!

Art. L. 112-2 du Code de la Propriété Intellectuelle 1994

"Sont considérés notamment comme œuvres de l'esprit au sens du présent code: …
13o «Les logiciels, y compris le matériel de conception préparatoire»; … "

Software is a pillar of Open Science

Software powers modern research



Proportion of publications in France published in 2022 that mention the use of code or software by discipline

French Open Science Monitor, Sources : MESR,

Over 20% of articles using software across all disciplines share it

2024 French Open Science Monitor

Source code is *special* (software is *not* data)

Software *evolves* over time

- projects may last decades
- the *development history* is key to its *understanding*

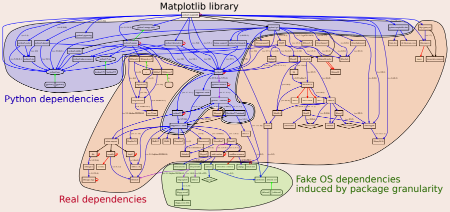
Source code is *special* (software is *not* data)

Software *evolves* over time

- projects may last decades
- the *development history* is key to its *understanding*

Complexity

- *millions* of lines of code
- large *web of dependencies*
  - easy to break, difficult to maintain
  - *research software* a thin top layer
- sophisticated *developer communities*



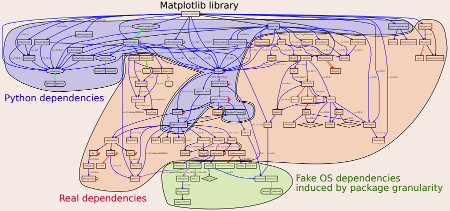
Source code is *special* (software is *not* data)

Software *evolves* over time

- projects may last decades
- the *development history* is key to its *understanding*

Complexity

- *millions* of lines of code
- large *web of dependencies*
  - easy to break, difficult to maintain
  - *research software* a thin top layer
- sophisticated *developer communities*



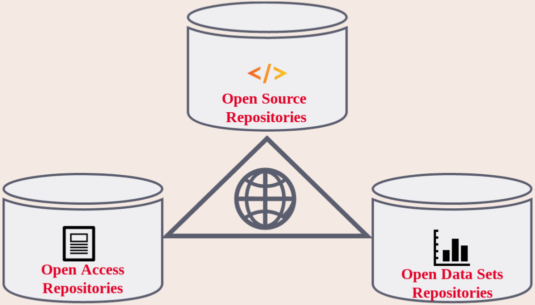
The human side

design, algorithm, code, test, documentation, community, funding

and so many more facets …

Software is a pillar of Open Science

Key pillar: software



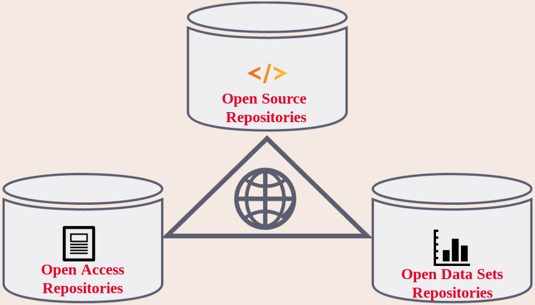
Links are important

Nota Bene

software may be a *tool*, a *research outcome* and a *research object*

Software is a pillar of Open Science

Key pillar: software



Links are important

Nota Bene

software may be a *tool*, a *research outcome* and a *research object*

access to the *source code* is essential!

Preserving (the history of) source code is necessary for *reproducibility*

Fundamental needs for software in Open Science (selection)

Archive

Research software artifacts must be properly archived

make sure we can *retrieve* them (*reproducibility*)

Fundamental needs for software in Open Science (selection)

Archive

Research software artifacts must be properly archived

make sure we can *retrieve* them (*reproducibility*)

Reference

Research software artifacts must be properly referenced

make sure we can *identify* them (*reproducibility*)

Fundamental needs for software in Open Science (selection)

Archive

Research software artifacts must be properly archived

make sure we can *retrieve* them (*reproducibility*)

Reference

Research software artifacts must be properly referenced

make sure we can *identify* them (*reproducibility*)

Describe

Research software artifacts must be properly described

make it easy to *discover* and *reuse* them (*visibility*)

Fundamental needs for software in Open Science (selection)

Archive

Research software artifacts must be properly archived

make sure we can *retrieve* them (*reproducibility*)

Reference

Research software artifacts must be properly referenced

make sure we can *identify* them (*reproducibility*)

Describe

Research software artifacts must be properly described

make it easy to *discover* and *reuse* them (*visibility*)

Cite/Credit

Research software artifacts must be properly cited *(not the same as referenced!)*

to give *credit* to authors (*evaluation*!)

Where is the source code?

Collaborative development platforms (aka "forges")

- BitBucket, GitLab(.com), GitHub, etc.
- support for version control, issues, etc.
- example:
  - https://depot.lipn.univ-paris13.fr/cosyverif/cosydraw
  - https://gitlab.inria.fr/gt-sw-citation/bibtex-sw-entry/

Where is the source code?

Collaborative development platforms (aka "forges")

- BitBucket, GitLab(.com), GitHub, etc.
- support for version control, issues, etc.
- example:
  - https://depot.lipn.univ-paris13.fr/cosyverif/cosydraw
  - https://gitlab.inria.fr/gt-sw-citation/bibtex-sw-entry/

Distribution platforms

- CTAN, CRAN, PyPi, Debian, etc.
- example: https://ctan.org/pkg/biblatex-software

Where is the source code?

Collaborative development platforms (aka "forges")

- BitBucket, GitLab(.com), GitHub, etc.
- support for version control, issues, etc.
- example:
  - https://depot.lipn.univ-paris13.fr/cosyverif/cosydraw
  - https://gitlab.inria.fr/gt-sw-citation/bibtex-sw-entry/

Distribution platforms

- CTAN, CRAN, PyPi, Debian, etc.
- example: https://ctan.org/pkg/biblatex-software

Archives

- Software Heritage
- example: archived version of biblatex-software

Archive and reference: some popular approaches that do not fit the bill

A - Since the ~~1970’s~~ 1990’s

.zip or .tar file on:

- ~~ftp server~~ (e.g. gnu)
- web page (example)
- document archive (+ DOI sample)

Archive and reference: some popular approaches that do not fit the bill

A - Since the ~~1970's~~ 1990's

.zip or .tar file on:

- ~~ftp server~~ (e.g. gnu)
- web page (example)
- document archive (+ DOI sample)

B - Since the 2000's

Rely on *software forges*

- institutional/project (e.g. example)
- free commercial ones: BitBucket, GitHub, GitLab, ... (e.g. imitator)

Archive and reference: some popular approaches that do not fit the bill

A - Since the ~~1970's~~ 1990's

.zip or .tar file on:

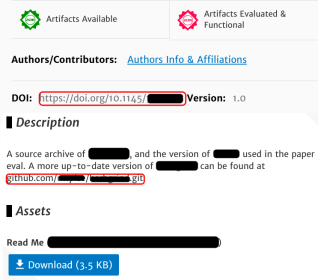
- ~~ftp server~~ (e.g. gnu)
- web page (example)
- document archive (+ DOI sample)

B - Since the 2000's

Rely on *software forges*

- institutional/project (e.g. example)
- free commercial ones: BitBucket, GitHub, GitLab, … (e.g. imitator)

C: a mix of the two



Authors/Contributors: Authors Info & Affiliations

DOI: https://doi.org/10.1145/[redacted] **Version:** 1.0

Description

A source archive of [redacted], and the version of [redacted] used in the paper eval. A more up-to-date version of [redacted] can be found at github.com/[redacted]/[redacted].git

Assets

Read Me [redacted])

Archive and reference: some popular approaches that do not fit the bill

A - Since the ~~1970's~~ 1990's

.zip or .tar file on:

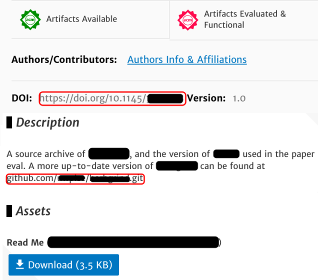
- ~~ftp server~~ (e.g. gnu)
- web page (example)
- document archive (+ DOI sample)

B - Since the 2000's

Rely on *software forges*

- institutional/project (e.g. example)
- free commercial ones: BitBucket, GitHub, GitLab, … (e.g. imitator)

C: a mix of the two



Artifacts Available

Artifacts Evaluated & Functional

Authors/Contributors: Authors Info & Affiliations

DOI: https://doi.org/10.1145/[redacted] **Version:** 1.0

Description

A source archive of [redacted], and the version of [redacted] used in the paper eval. A more up-to-date version of [redacted] can be found at github.com/[redacted]/[redacted].git

Assets

Read Me [redacted])

Can get no satisfaction…

A *Poor user experience*

B *No preservation guarantee*

C Can do *so much* better

Forges are *not* archives!

2015: the first big bad news

Google Code and Gitorious.org shutdown: ~1M endangered repositories

- broken links in the web of knowledge

Forges are *not* archives!

2015: the first big bad news

Google Code and Gitorious.org shutdown: ~1M endangered repositories

- broken links in the web of knowledge

Big bad news keep coming in

- summer 2019: BitBucket announces Mercurial VCS sunset
- july 2020: BitBucket erases *250.000+* repositories (including research software)
- summer 2022: GitLab.com considers erasing all projects that are inactive for a year

Forges are *not* archives!

2015: the first big bad news

Google Code and Gitorious.org shutdown: ~1M endangered repositories

- broken links in the web of knowledge

Big bad news keep coming in

- summer 2019: BitBucket announces Mercurial VCS sunset
- july 2020: BitBucket erases *250.000+* repositories (including research software)
- summer 2022: GitLab.com considers erasing all projects that are inactive for a year

In Academia too!

- 2021: Inria's old gforge is unplugged… breaks the Opam build chain for OCaml

Forges are *not* archives!

2015: the first big bad news

Google Code and Gitorious.org shutdown: ~1M endangered repositories

- broken links in the web of knowledge

Big bad news keep coming in

- summer 2019: BitBucket announces Mercurial VCS sunset
- july 2020: BitBucket erases *250.000+* repositories (including research software)
- summer 2022: GitLab.com considers erasing all projects that are inactive for a year

In Academia too!

- 2021: Inria's old gforge is unplugged... breaks the Opam build chain for OCaml

We need a universal archive of software source code:

Forges are *not* archives!

2015: the first big bad news

Google Code and Gitorious.org shutdown: ~1M endangered repositories

- broken links in the web of knowledge

Big bad news keep coming in

- summer 2019: BitBucket announces Mercurial VCS sunset
- july 2020: BitBucket erases *250.000+* repositories (including research software)
- summer 2022: GitLab.com considers erasing all projects that are inactive for a year

In Academia too!

- 2021: Inria's old gforge is unplugged… breaks the Opam build chain for OCaml

We need a universal archive of software source code: now we have one!

Outline

1. Open Science & Software
2. Software Heritage for Open Science and Reproducibility
3. Quick Demo !
4. Call to action
5. Conclusion

Software Heritage in a nutshell

www.softwareheritage.org

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Collect, preserve and share *all* software source code

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

Software Heritage in a nutshell

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Collect, preserve and share *all* software source code

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

Reference catalog

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

find and reference all software source code

Software Heritage in a nutshell

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Collect, preserve and share *all* software source code

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

Reference catalog

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

find and reference all software source code

Universal archive

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

preserve and share all software source code

Software Heritage in a nutshell

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Collect, preserve and share *all* software source code

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

Reference catalog

find and reference all software source code

Universal archive

preserve and share all software source code

Research infrastructure

enable analysis of all software source code

An international, non profit initiative built for the long term

Sharing the vision

UNESCO

United Nations
Educational, Scientific and
Cultural Organization

IEEE Software
GitLab
ENGINEERING
eclipse
OW2
fsfe
nlnet FOUNDATION
SIGSOFT
Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE
INAPI
THE LINUX FOUNDATION
ADULLACT
INFORMATICS EUROPE
Computer History Museum
software freedom conservancy
Software Freedom Law Center
AdaCore The GNAT Pro Company
gandi.net
TDF
FREE SOFTWARE FOUNDATION
creative commons
SIF Société Informatique de France
open source initiative
openinventionnetwork

And many more ...
www.softwareheritage.org/support/testimonials

An international, non profit initiative built for the long term

Sharing the vision

UNESCO
United Nations
Educational, Scientific and
Cultural Organization

IEEE Software · GitLab · ENGINEERING · eclipse
OW2 · fsfe · nlnet FOUNDATION · SIGSOFT · Liberté • Égalité • Fraternité RÉPUBLIQUE FRANÇAISE · INAPI
THE LINUX FOUNDATION
ADULLACT · INFORMATICS EUROPE · Computer History Museum · software freedom conservancy
Software Freedom Law Center · AdaCore The GNAT Pro Company · gandi.net · TDF
FREE SOFTWARE FOUNDATION
creative commons · SIF Société Informatique de France · open source initiative · openinventionnetwork

And many more …
www.softwareheritage.org/support/testimonials

Donors, members, sponsors

Inria

Diamond sponsor
cea

Platinum sponsors
cnrs · intel · MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE · Microsoft · HUAWEI

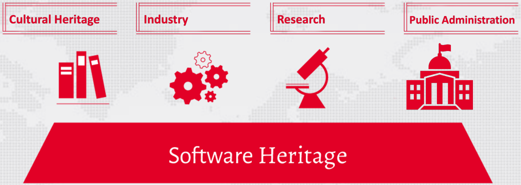
Gold sponsors
Hugging Face · openinventionnetwork · servicenow
SOCIETE GENERALE · SORBONNE UNIVERSITÉ · Université Paris Cité

Silver sponsors
AdaCore · MINISTÈRE DES ARMÉES · DGA · RÉPUBLIQUE FRANÇAISE · GitHub · Google · UNIVERSITÀ DI PISA

Bronze sponsors
SCANOSS · SCUOLA NORMALE SUPERIORE · ALMA MATER STUDIORUM A.D. 1088 · UNIVERSITÉ DE LORRAINE

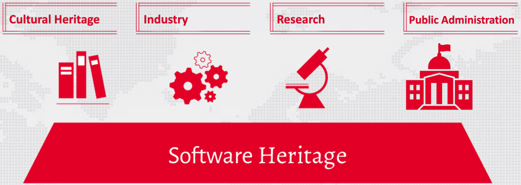
The largest software archive, a shared infrastructure

One infrastructure
open and shared

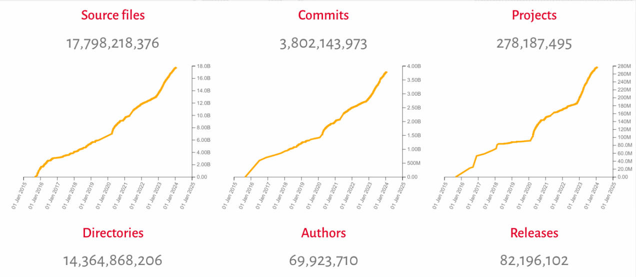


The largest software archive, a shared infrastructure

One infrastructure
open and shared


Cultural Heritage
Industry
Research
Public Administration
Software Heritage

The largest archive ever built


Source files
17,798,218,376

Commits
3,802,143,973

Projects
278,187,495

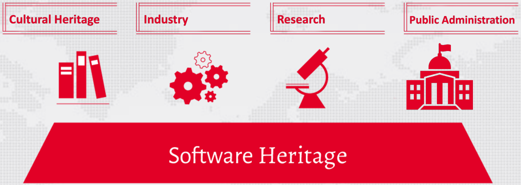
Directories
14,364,868,206

Authors
69,923,710

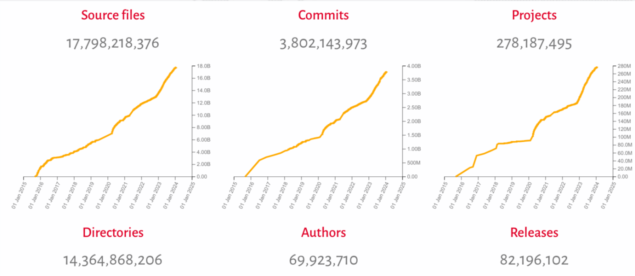
Releases
82,196,102

The largest software archive, a shared infrastructure

One infrastructure
open and shared

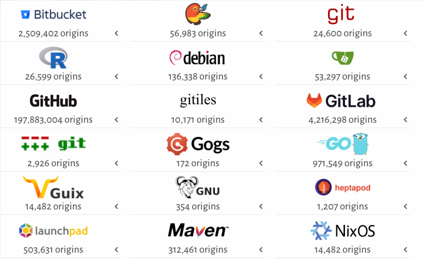


The largest archive ever built



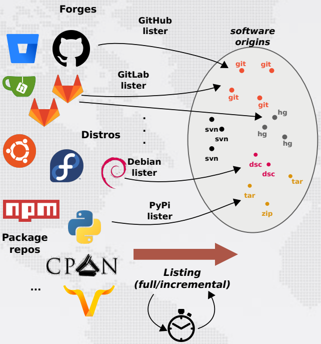
Source files	Commits	Projects
17,798,218,376	3,802,143,973	278,187,495

Directories	Authors	Releases
14,364,868,206	69,923,710	82,196,102

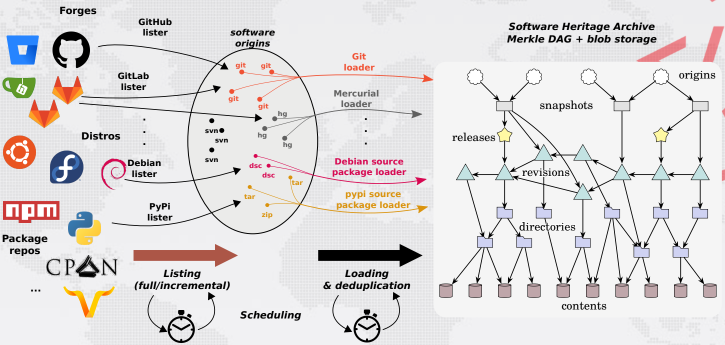


Bitbucket 2,509,402 origins	56,983 origins	git 24,600 origins
R 26,599 origins	debian 136,338 origins	53,297 origins
GitHub 197,883,004 origins	gitiles 10,171 origins	GitLab 4,216,298 origins
git 2,926 origins	Gogs 172 origins	GO 971,549 origins
Guix 14,482 origins	GNU 354 origins	heptapod 1,207 origins
launchpad 503,631 origins	Maven 312,461 origins	NixOS 14,482 origins

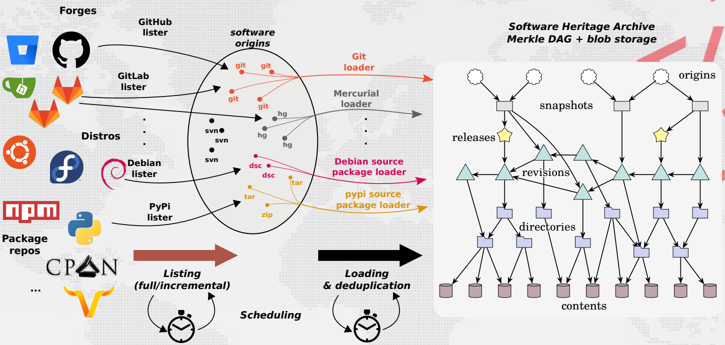
Software Heritage: a *radically different* approach to archiving



Software Heritage: a *radically different* approach to archiving



Software Heritage: a *radically different* approach to archiving



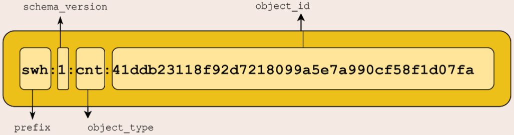
Global development history permanently archived in a uniform data model

- over 20 billion unique source files from over 300 million software projects
- ~2PB (compressed) blobs, ~50 B nodes, ~700 B edges

Software Heritage is *radically different*, cont'd

Software Hash Identifiers (SWHID) see swhid.org

50+B intrinsic, decentralised, cryptographically strong identifiers, SWHIDs

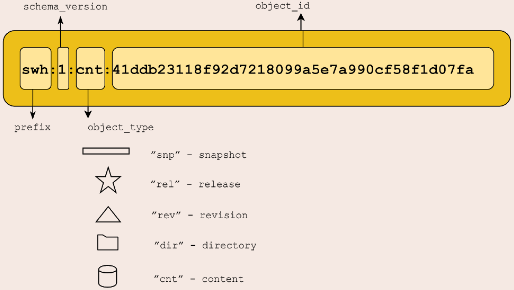


Software Heritage is *radically different*, cont'd

Software Hash Identifiers (SWHID)

see swhid.org

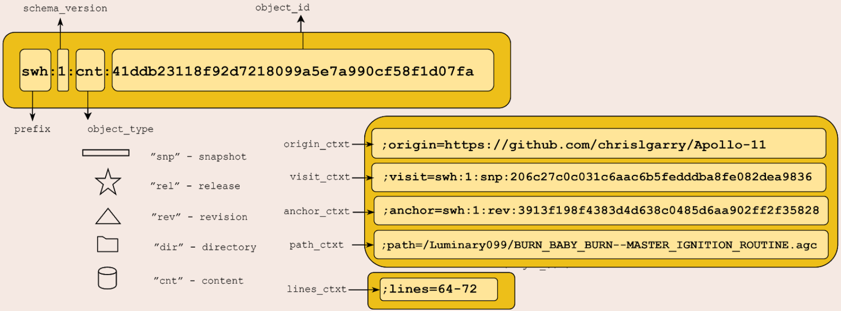
50+B intrinsic, decentralised, cryptographically strong identifiers, SWHIDs



Software Heritage is *radically different*, cont'd

Software Hash Identifiers (SWHID) see swhid.org

50+B intrinsic, decentralised, cryptographically strong identifiers, SWHIDs



schema_version

object_id

swh:1:cnt:41ddb23118f92d7218099a5e7a990cf58f1d07fa

prefix

object_type

"snp" - snapshot

"rel" - release

"rev" - revision

"dir" - directory

"cnt" - content

origin_ctxt → ;origin=https://github.com/chrislgarry/Apollo-11

visit_ctxt → ;visit=swh:1:snp:206c27c0c031c6aac6b5fedddba8fe082dea9836

anchor_ctxt → ;anchor=swh:1:rev:3913f198f4383d4d638c0485d6aa902ff2f35828

path_ctxt → ;path=/Luminary099/BURN_BABY_BURN--MASTER_IGNITION_ROUTINE.agc

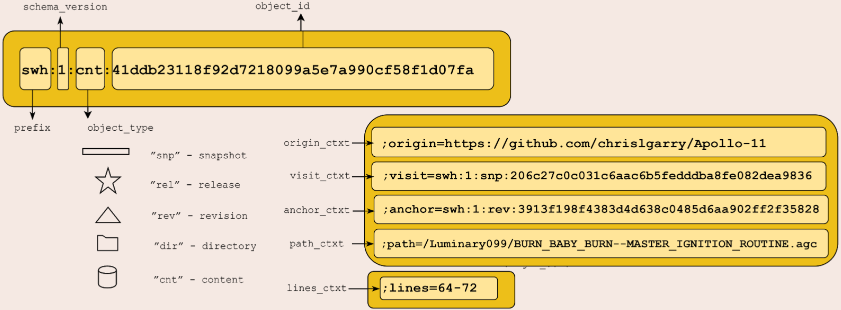
lines_ctxt → ;lines=64-72

Software Heritage is *radically different*, cont'd

Software Hash Identifiers (SWHID)

see swhid.org

50+B intrinsic, decentralised, cryptographically strong identifiers, SWHIDs



schema_version

object_id

swh:1:cnt:41ddb23118f92d7218099a5e7a990cf58f1d07fa

prefix

object_type

▭ "snp" - snapshot

☆ "rel" - release

△ "rev" - revision

🗀 "dir" - directory

⛁ "cnt" - content

origin_ctxt → ;origin=https://github.com/chrislgarry/Apollo-11

visit_ctxt → ;visit=swh:1:snp:206c27c0c031c6aac6b5fedddba8fe082dea9836

anchor_ctxt → ;anchor=swh:1:rev:3913f198f4383d4d638c0485d6aa902ff2f35828

path_ctxt → ;path=/Luminary099/BURN_BABY_BURN--MASTER_IGNITION_ROUTINE.agc

lines_ctxt → ;lines=64-72

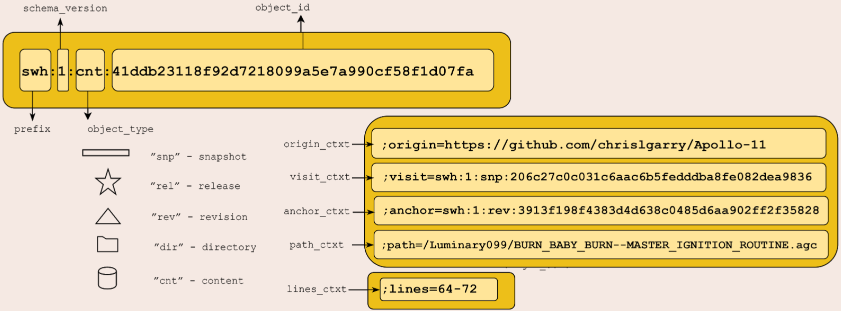
In SPDX 2.2; IANA registered "swh:"; WikiData P6138; ISO standard

Software Heritage is *radically different*, cont'd

Software Hash Identifiers (SWHID)

see swhid.org

50+B intrinsic, decentralised, cryptographically strong identifiers, SWHIDs



schema_version

object_id

swh:1:cnt:41ddb23118f92d7218099a5e7a990cf58f1d07fa

prefix

object_type

"snp" - snapshot

"rel" - release

"rev" - revision

"dir" - directory

"cnt" - content

origin_ctxt ;origin=https://github.com/chrislgarry/Apollo-11

visit_ctxt ;visit=swh:1:snp:206c27c0c031c6aac6b5fedddba8fe082dea9836

anchor_ctxt ;anchor=swh:1:rev:3913f198f4383d4d638c0485d6aa902ff2f35828

path_ctxt ;path=/Luminary099/BURN_BABY_BURN--MASTER_IGNITION_ROUTINE.agc

lines_ctxt ;lines=64-72

In SPDX 2.2; IANA registered "swh:"; WikiData P6138; ISO standard

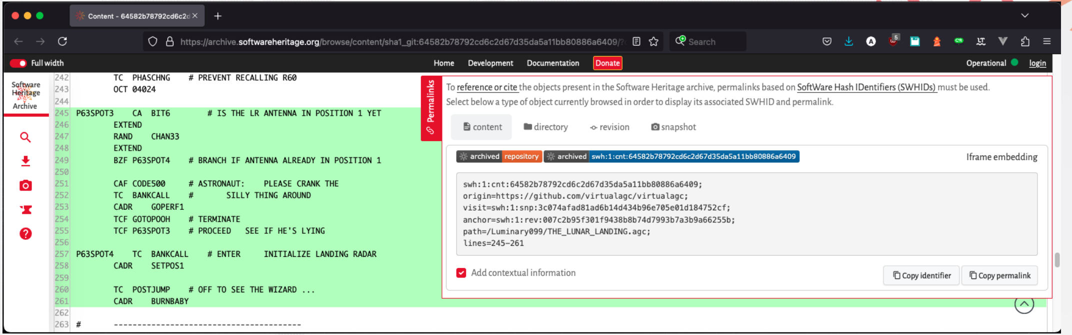
Full fledged *source code references* for traceability, integrity and reproducibility

Examples: Apollo 11 AGC, Quake III rsqrt; Guidelines available: HOWTO and ICMS 2020

Software Heritage is *radically different*, cont'd

A quick tour as a user

- designed for source code: Browse (e.g. Apollo 11 excerpt) like on a developer platform, not a document archive!



Content - 64582b78792cd6c2d
https://archive.softwareheritage.org/browse/content/sha1_git:64582b78792cd6c2d67d35da5a11bb80886a6409/
Search
Full width
Home
Development
Documentation
Donate
Operational
login
Software Heritage Archive
Permalinks
content
directory
revision
snapshot
archived repository
archived swh:1:cnt:64582b78792cd6c2d67d35da5a11bb80886a6409
Iframe embedding
Copy identifier
Copy permalink

```
242          TC   PHASCHNG    # PREVENT RECALLING R60
243          OCT 04024
244
245 P63SPOT3    CA  BIT6        # IS THE LR ANTENNA IN POSITION 1 YET
246          EXTEND
247          RAND    CHAN33
248          EXTEND
249          BZF P63SPOT4    # BRANCH IF ANTENNA ALREADY IN POSITION 1
250
251          CAF CODE500     # ASTRONAUT:    PLEASE CRANK THE
252          TC  BANKCALL    #       SILLY THING AROUND
253          CADR    GOPERF1
254          TCF GOTOPOOH    # TERMINATE
255          TCF P63SPOT3    # PROCEED   SEE IF HE'S LYING
256
257 P63SPOT4    TC  BANKCALL    # ENTER     INITIALIZE LANDING RADAR
258          CADR    SETPOS1
259
260          TC  POSTJUMP    # OFF TO SEE THE WIZARD ...
261          CADR    BURNBABY
262
263 #        ----------------------------------------
```

To reference or cite the objects present in the Software Heritage archive, permalinks based on SoftWare Hash IDentifiers (SWHIDs) must be used.

Select below a type of object currently browsed in order to display its associated SWHID and permalink.

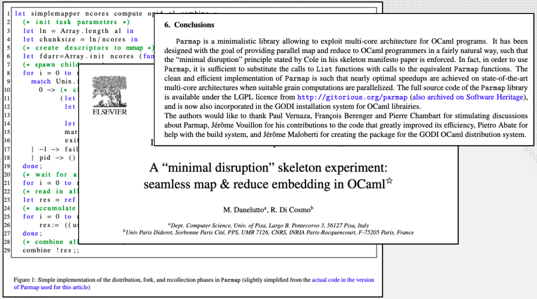
```
swh:1:cnt:64582b78792cd6c2d67d35da5a11bb80886a6409;
origin=https://github.com/virtualagc/virtualagc;
visit=swh:1:snp:3c074afad81ad6b14d434b96e705e01d184752cf;
anchor=swh:1:rev:007c2b95f301f9438b8b74d7993b7a3b9a66255b;
path=/Luminary099/THE_LUNAR_LANDING.agc;
lines=245-261
```

☑ Add contextual information

Software Heritage is *radically different*, cont'd

A quick tour as a user

- reference source code: all granularities, using SWHIDs (full specification available online)
    - SWHIDs *guarantee integrity* like in *blockchains*



```
let simplemapper ncores compute op
  (* init task parameters *)
  let ln = Array.length al in
  let chunksize = ln/ncores in
  (* create descriptors to mmap *)
  let fdarr=Array.init ncores (fun
  (* spawn child
  for i = 0 to 
    match Unix.
      0 -> (* cl
           (let
            let

            let
            mar
            exit
    | -1 -> fail
    | pid -> ()
  done;
  (* wait for a
  for i = 0 to 
  (* read in all
  let res = ref
  (* accumulate
  for i = 0 to 
      res:= ((u
  done;
  (* combine all
  combine !res;;
```

Figure 1: Simple implementation of the distribution, fork, and recollection phases in Parmap (slightly simplified from the actual code in the version of Parmap used for this article)

ELSEVIER

A "minimal disruption" skeleton experiment: seamless map & reduce embedding in OCaml☆

M. Danelutto[a], R. Di Cosmo[b]

[a]Dept. Computer Science, Univ. of Pisa, Largo B. Pontecorvo 3, 56127 Pisa, Italy
[b]Univ Paris Diderot, Sorbonne Paris Cité, PPS, UMR 7126, CNRS, INRIA Paris-Rocquencourt, F-75205 Paris, France

6. Conclusions

Parmap is a minimalistic library allowing to exploit multi-core architecture for OCaml programs. It has been designed with the goal of providing parallel map and reduce to OCaml programmers in a fairly natural way, such that the "minimal disruption" principle stated by Cole in his skeleton manifesto paper is enforced. In fact, in order to use Parmap, it is sufficient to substitute the calls to List functions with calls to the equivalent Parmap functions. The clean and efficient implementation of Parmap is such that nearly optimal speedups are achieved on state-of-the-art multi-core architectures when suitable grain computations are parallelized. The full source code of the Parmap library is available under the LGPL licence from http://gitorious.org/parmap (also archived on Software Heritage), and is now also incorporated in the GODI installation system for OCaml librairies.

The authors would like to thank Paul Vernaza, François Berenger and Pierre Chambart for stimulating discussions about Parmap, Jérôme Vouillon for his contributions to the code that greatly improved its efficiency, Pietro Abate for help with the build system, and Jérôme Maloberti for creating the package for the GODI OCaml distribution system.

Figure: Compare Fig. 1 and conclusions in the 2012 version and the updated version

Software Heritage is *radically different*, cont'd

Getting software archived

- automated harvesting: over 290 million software origins, your researchers' work may already be there (actually, here)!

Software Heritage is *radically different*, cont'd

Getting software archived

- automated harvesting: over 290 million software origins, your researchers' work may already be there (actually, here)!
- universal archive: *all* source code from *all* platforms (BitBucket, GitHub, GitLab, your own forge, etc.)
  - trigger archival of *any code* in one click with the updateswh browser extension
  - use webhooks to automatically archive *your code* (a GitHub action is available too)
  - journals, libraries, open access portals may *deposit sourcecode and metadata*
    - Example article from IPOL
    - Example article from eLife

Outline

1. Open Science & Software
2. Software Heritage for Open Science and Reproducibility
3. Quick Demo !
4. Call to action
5. Conclusion

A walkthrough

- Browse (e.g. Imitator [excerpt], your work may be already there !)
- Trigger archival, use the updateswh browser extension, configure the webhooks
- Get and use SWHIDs (full specification available online)
- Cite software with biblatex-software package from CTAN
  - Overleaf ACMART template available
- Example in journals: article from IPOL
- Example with adt2amas: code source, archive in SWH, curated deposit in HAL
- Extracting all the software products for Inria, for CNRS, for CNES, for LIRMM or for Rémi Gribonval using HalTools
- Curated deposit in SWH via HAL, see for example: LinBox, SLALOM, Givaro, NS2DDV, SumGra, Coq proof, . . .

An example of long term reproducibility for HPC

(re)create fully reproducible binaries from source... https://guix.gnu.org/

- functional package manager
- bit by bit reproductibility
- *from the source code*

An example of long term reproducibility for HPC

(re)create fully reproducible binaries from source… https://guix.gnu.org/

- functional package manager
- bit by bit reproductibility
- *from the source code*

… with a focus on HPC https://hpc.guix.info/

GuixHPC
Reproducible software deployment for high-performance computing.

- environment control
- support cluster deployment
- *from the source code*

An example of long term reproducibility for HPC

(re)create fully reproducible binaries from source… https://guix.gnu.org/

Guix

- functional package manager
- bit by bit reproductibility
- *from the source code*

… with a focus on HPC https://hpc.guix.info/

GuixHPC

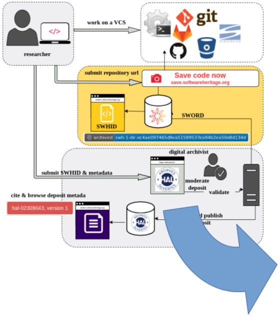
Reproducible software deployment for high-performance computing.

- environment control
- support cluster deployment
- *from the source code*

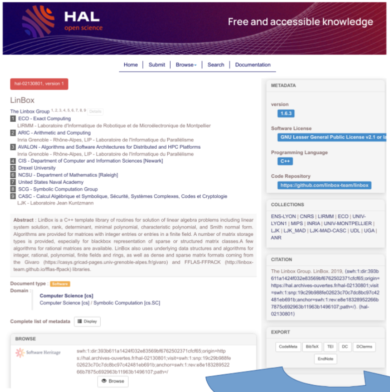
connection with Software Heritage

- source code *archival and identification* for `guix` and `nix`
- automatic fallback for missing sources (see experience report)

HAL and Software Heritage: building a curated software catalog



https://hal.archives-ouvertes.fr/hal-02130801


HAL
open science
Free and accessible knowledge
Home | Submit | Browse | Search | Documentation

hal-02130801, version 1

LinBox

The Linbox Group 1, 2, 3, 4, 5, 6, 7, 8, 9 Details

1. ECO - Exact Computing
   LIRMM - Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier
2. ARIC - Arithmetic and Computing
   Inria Grenoble - Rhône-Alpes, LIP - Laboratoire de l'Informatique du Parallélisme
3. AVALON - Algorithms and Software Architectures for Distributed and HPC Platforms
   Inria Grenoble - Rhône-Alpes, LIP - Laboratoire de l'Informatique du Parallélisme
4. CIS - Department of Computer and Information Sciences [Newark]
5. Drexel University
6. NCSU - Department of Mathematics [Raleigh]
7. United States Naval Academy
8. SCG - Symbolic Computation Group
9. CASC - Calcul Algébrique et Symbolique, Sécurité, Systèmes Complexes, Codes et Cryptologie
   LJK - Laboratoire Jean Kuntzmann

Abstract : LinBox is a C++ template library of routines for solution of linear algebra problems including linear system solution, rank, determinant, minimal polynomial, characteristic polynomial, and Smith normal form. Algorithms are provided for matrices with integer entries or entries in a finite field. A number of matrix storage types is provided, especially for blackbox representation of sparse or structured matrix classes.A few algorithms for rational matrices are available. LinBox also uses underlying data structures and algorithms for integer, rational, polynomial, finite fields and rings, as well as dense and sparse matrix formats coming from the Givaro (https://casys.gricad-pages.univ-grenoble-alpes.fr/givaro) and FFLAS-FFPACK (http://linbox-team.github.io/fflas-ffpack) libraries.

Document type : Software

Domain :
Computer Science [cs]
Computer Science [cs] / Symbolic Computation [cs.SC]

Complete list of metadata Display

BROWSE

Software Heritage swh:1:dir:393b611a1424f032e83569bf6762502371cfcf65;origin=https://hal.archives-ouvertes.fr/hal-02130801;visit=swh:1:snp:19c29b988fe02623c70c7dc8bc97c42481eb691b;anchor=swh:1:rev:e8e18328952266b7875c692963b11963b1496107;path=/

Browse

METADATA

version
1.6.3

Software License
GNU Lesser General Public License v2.1 or la

Programming Language
C++

Code Repository
https://github.com/linbox-team/linbox

COLLECTIONS

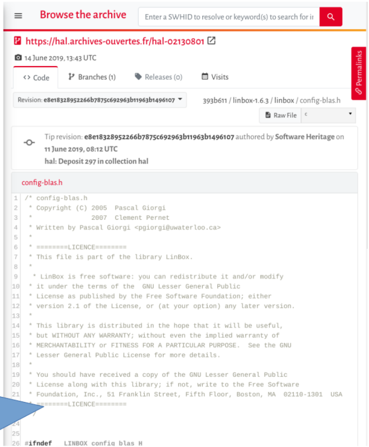
ENS-LYON | CNRS | LIRMM | ECO | UNIV-LYON1 | MIPS | INRIA | UNIV-MONTPELLIER | LJK | LJK_MAD | LJK-MAD-CASC | UDL | UGA | ANR

CITATION

The Linbox Group. LinBox. 2019, (swh:1:dir:393b611a1424f032e83569bf6762502371cfcf65;origin=https://hal.archives-ouvertes.fr/hal-02130801;visit=swh:1:snp:19c29b988fe02623c70c7dc8bc97c42481eb691b;anchor=swh:1:rev:e8e18328952266b7875c692963b11963b1496107;path=/). (hal-02130801)

EXPORT

CodeMeta | BibTeX | TEI | DC | DCterms | EndNote



https://hal.archives-ouvertes.fr/hal-02130801

14 June 2019, 13:43 UTC

Revision: e8e18328952266b7875c692963b11963b1496107

393b611 / linbox-1.6.3 / linbox / config-blas.h

Tip revision: e8e18328952266b7875c692963b11963b1496107 authored by Software Heritage on 11 June 2019, 08:12 UTC
hal: Deposit 297 in collection hal

config-blas.h

```
/* config-blas.h
 * Copyright (C) 2005  Pascal Giorgi
 *                2007  Clement Pernet
 * Written by Pascal Giorgi <pgiorgi@uwaterloo.ca>
 *
 * ========LICENCE========
 * This file is part of the library LinBox.
 *
  * LinBox is free software: you can redistribute it and/or modify
 * it under the terms of the  GNU Lesser General Public
 * License as published by the Free Software Foundation; either
 * version 2.1 of the License, or (at your option) any later version.
 *
 * This library is distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
 * Lesser General Public License for more details.
 *
 * You should have received a copy of the GNU Lesser General Public
 * License along with this library; if not, write to the Free Software
 * Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA
 * ========LICENCE========
 */


#ifndef   LINBOX config blas H
```

swh:1:dir:393b611a1424f032e83569bf6762502371cfcf65

with minimal user overhead!

Outline

1. Open Science & Software

2. Software Heritage for Open Science and Reproducibility

3. Quick Demo !

4. Call to action

5. Conclusion

Call to action: best practices for ARDC are available. . . today!

Archiving and referencing

For all source code used in research (*yes, even small scripts!*)

- ensure it is archived in Software Heritage (see save code now)
- get the proper SWHID for your software (see detailed HOWTO)
- add it to research articles for reproducibility (see detailed HOWTO)

Call to action: best practices for ARDC are available… today!

Archiving and referencing

For all source code used in research (*yes, even small scripts!*)

- ensure it is archived in Software Heritage (see save code now)
- get the proper SWHID for your software (see detailed HOWTO)
- add it to research articles for reproducibility (see detailed HOWTO)

Describing and Citing/Crediting

For software you want to put forward (*mention in your CV, reports, etc., get citations and credit for it*), do the following extra steps:

- add codemeta.json with description (see the codemeta generator)
- reference in the HAL portal (french partners, see online HAL documentation)
- cite software using the biblatex-software package (in CTAN and TeXLive)

Call to action: best practices for ARDC are available... today!

Archiving and referencing

For all source code used in research (*yes, even small scripts!*)

- ensure it is archived in Software Heritage (see save code now)
- get the proper SWHID for your software (see detailed HOWTO)
- add it to research articles for reproducibility (see detailed HOWTO)

Describing and Citing/Crediting

For software you want to put forward (*mention in your CV, reports, etc., get citations and credit for it*), do the following extra steps:

- add codemeta.json with description (see the codemeta generator)
- reference in the HAL portal (french partners, see online HAL documentation)
- cite software using the biblatex-software package (in CTAN and TeXLive)

- train students, colleagues
- engage journals, conferences, learned societies

Outline

1. Open Science & Software
2. Software Heritage for Open Science and Reproducibility
3. Quick Demo !
4. Call to action
5. Conclusion

A rally flag for a grand vision

Bring together academia, industry, governments, communities

"to build a reference, global infrastructure for open and better software"

A rally flag for a grand vision

Bring together academia, industry, governments, communities

"to build a reference, global infrastructure for open and better software"

Software Heritage is the first brick …

- vendor neutral
- open source
- a worldwide initiative
- a long term initiative

A rally flag for a grand vision

Bring together academia, industry, governments, communities

"to build a reference, global infrastructure for open and better software"

Software Heritage is the first brick ...

- vendor neutral
- open source
- a worldwide initiative
- a long term initiative

... that will enable

- archival, reference, integrity
- qualification, sharing and reuse
- a global software knowledge base
- test and deploy world class tooling

A rally flag for a grand vision

Bring together academia, industry, governments, communities

"to build a reference, global infrastructure for open and better software"

Software Heritage is the first brick …

- vendor neutral
- open source
- a worldwide initiative
- a long term initiative

… that will enable

- archival, reference, integrity
- qualification, sharing and reuse
- a global software knowledge base
- test and deploy world class tooling

A lot more is needed

Software Heritage can be the *catalyser* of a way bigger undertaking

A rally flag for a grand vision

Bring together academia, industry, governments, communities

"to build a reference, global infrastructure for open and better software"

Software Heritage is the first brick …

- vendor neutral
- open source
- a worldwide initiative
- a long term initiative

… that will enable

- archival, reference, integrity
- qualification, sharing and reuse
- a global software knowledge base
- test and deploy world class tooling

A lot more is needed

Software Heritage can be the *catalyser* of a way bigger undertaking

You can help!

use, disseminate, contribute, build&adapt research tools, …

Join a growing and active community

Team

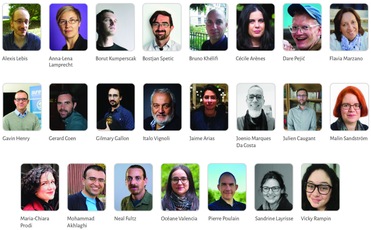
Software Heritage
THE GREAT LIBRARY OF SOURCE CODE
We are building the universal software archive
softwareheritage.org
a project launched by Inria

Join a growing and active community

Team

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE
We are building the universal software archive
softwareheritage.org
a project launched by Inria

Ambassadors

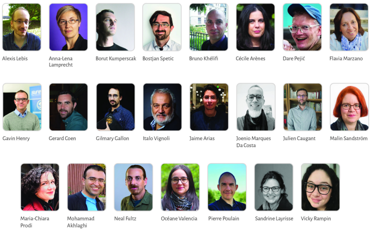


Join a growing and active community

Team

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE
We are building the universal software archive
softwareheritage.org
a project launched by Inria

Ambassadors

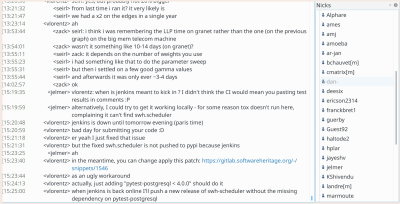


Alexis Lebis, Anna-Lena Lamprecht, Borut Kumperscak, Bostjan Spetic, Bruno Khélifi, Cécile Arènes, Dare Pejić, Flavia Marzano

Gavin Henry, Gerard Coen, Gilmary Gallon, Italo Vignoli, Jaime Arias, Joenio Marques Da Costa, Julien Caugant, Malin Sandström

Maria-Chiara Prodi, Mohammad Akhlaghi, Neal Fultz, Océane Valencia, Pierre Poulain, Sandrine Layrisse, Vicky Rampin

Contributors to the platform



```
[13:20:50  <vlorentz> seirl: yes, but probably not 20% bigger
[13:21:32      <seirl> from last time i ran it? it very likely is
[13:21:47      <seirl> we had a x2 on the edges in a single year
[13:23:14  <vlorentz> ah
[13:53:44       <zack> seirl: i think i was remembering the LLP time on granet rather than the one (on the previous
                       graph) on the big mem telecom machine
[13:54:01       <zack> wasn't it something like 10-14 days (on granet)?
[13:55:11      <seirl> zack: it depends on the number of weights you use
[13:55:23      <seirl> i had something like that to do the parameter sweep
[13:55:31      <seirl> but then i settled on a few good gamma values
[13:55:44      <seirl> and afterwards it was only ever ~3-4 days
[14:02:57       <zack> ok
[15:19:35     <jelmer> vlorentz: when is jenkins meant to kick in ? I didn't think the CI would mean you pasting test
                       results in comments :P
[15:19:59     <jelmer> alternatively, I could try to get it working locally - for some reason tox doesn't run here,
                       complaining it can't find swh.scheduler
[15:20:48  <vlorentz> jenkins is down until tomorrow evening (paris time)
[15:20:59  <vlorentz> bad day for submitting your code :D
[15:21:18  <vlorentz> er yeah I just fixed that issue
[15:21:31  <vlorentz> but the fixed swh.scheduler is not pushed to pypi because jenkins
[15:23:25     <jelmer> ah
[15:23:40  <vlorentz> in the meantime, you can change apply this patch: https://gitlab.softwareheritage.org/-/
                       snippets/1546
[15:23:44  <vlorentz> as an ugly workaround
[15:24:13  <vlorentz> actually, just adding "pytest-postgresql < 4.0.0" should do it
[15:25:00  <vlorentz> when jenkins is back online I'll push a new release of swh-scheduler without the missing
                       dependency on pytest-postgresql
```

Nicks

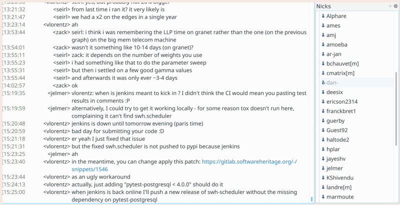
- Alphare
- ames
- amj
- amoeba
- ar-jan
- bchauvet[m]
- cmatrix[m]
- dan-
- deesix
- ericson2314
- franckbret1
- guerby
- Guest92
- haltode2
- hplar
- jayeshv
- jelmer
- KShivendu
- landre[m]
- marmoute

Join a growing and active community

Team

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE
We are building the universal software archive
softwareheritage.org
a project launched by Inria

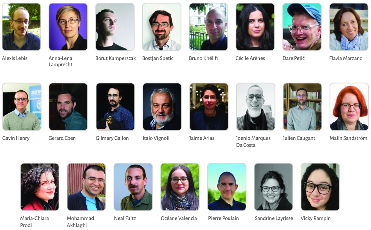
Contributors to the platform



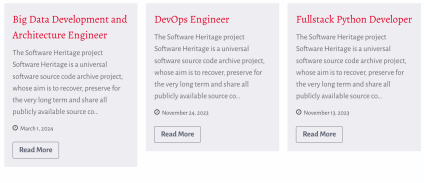
```
[13:20:56   <vlorentz> seirl: yes, but probably not 20% bigger
[13:21:32       <seirl> from last time i ran it? it very likely is
[13:21:47       <seirl> we had a x2 on the edges in a single year
[13:23:14   <vlorentz> ah
[13:53:44       <zack> seirl: i think i was remembering the LLP time on granet rather than the one (on the previous
                       graph) on the big mem telecom machine
[13:54:01       <zack> wasn't it something like 10-14 days (on granet)?
[13:55:11       <seirl> zack: it depends on the number of weights you use
[13:55:23       <seirl> i had something like that to do the parameter sweep
[13:55:31       <seirl> but then i settled on a few good gamma values
[13:55:44       <seirl> and afterwards it was only ever ~3-4 days
[14:02:57       <zack> ok
[15:19:35     <jelmer> vlorentz: when is jenkins meant to kick in ? I didn't think the CI would mean you pasting test
                       results in comments :P
[15:19:59     <jelmer> alternatively, I could try to get it working locally - for some reason tox doesn't run here,
                       complaining it can't find swh.scheduler
[15:20:48   <vlorentz> jenkins is down until tomorrow evening (paris time)
[15:20:59   <vlorentz> bad day for submitting your code :D
[15:21:18   <vlorentz> er yeah I just fixed that issue
[15:21:31   <vlorentz> but the fixed swh.scheduler is not pushed to pypi because jenkins
[15:23:25     <jelmer> ah
[15:23:40   <vlorentz> in the meantime, you can change apply this patch: https://gitlab.softwareheritage.org/-/
                       snippets/1546
[15:23:44   <vlorentz> as an ugly workaround
[15:24:13   <vlorentz> actually, just adding "pytest-postgresql < 4.0.0" should do it
[15:25:00   <vlorentz> when jenkins is back online I'll push a new release of swh-scheduler without the missing
                       dependency on pytest-postgresql
```

Nicks
- Alphare
- ames
- amj
- amoeba
- ar-jan
- bchauvet[m]
- cmatrix[m]
- dan-
- deesix
- ericson2314
- franckbret1
- guerby
- Guest92
- haltode2
- hplar
- jayeshv
- jelmer
- KShivendu
- landre[m]
- marmoute

Ambassadors



Work with us!



Big Data Development and Architecture Engineer

The Software Heritage project Software Heritage is a universal software source code archive project, whose aim is to recover, preserve for the very long term and share all publicly available source co...

March 1, 2024

Read More

DevOps Engineer

The Software Heritage project Software Heritage is a universal software source code archive project, whose aim is to recover, preserve for the very long term and share all publicly available source co...

November 24, 2023

Read More

Fullstack Python Developer

The Software Heritage project Software Heritage is a universal software source code archive project, whose aim is to recover, preserve for the very long term and share all publicly available source co...

November 13, 2023

Read More

https://softwareheritage.org/jobs/

Report and videos

Annual report

Software Heritage

Annual Report 2023

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Collecting, preserving and sharing software source code since 2015

Report and videos

Annual report

Software Heritage

Annual Report 2023

Software Heritage THE GREAT LIBRARY OF SOURCE CODE

Collecting, preserving and sharing software source code since 2015

5 years in 5 minutes Link

Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Report and videos

Annual report

Software Heritage

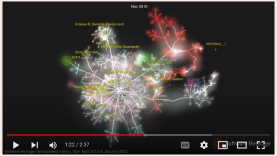
Annual Report 2023

Software Heritage THE GREAT LIBRARY OF SOURCE CODE

Collecting, preserving and sharing software source code since 2015

5 years in 5 minutes Link

Evolution of our codebase Link


Nov 2019
1:22 / 2:37
Software Heritage development history, from April 2015 to January 2023

it’s a long road, but together we can make it

Thank you

Credits

This presentation reuses material from Roberto di Cosmo's presentations.