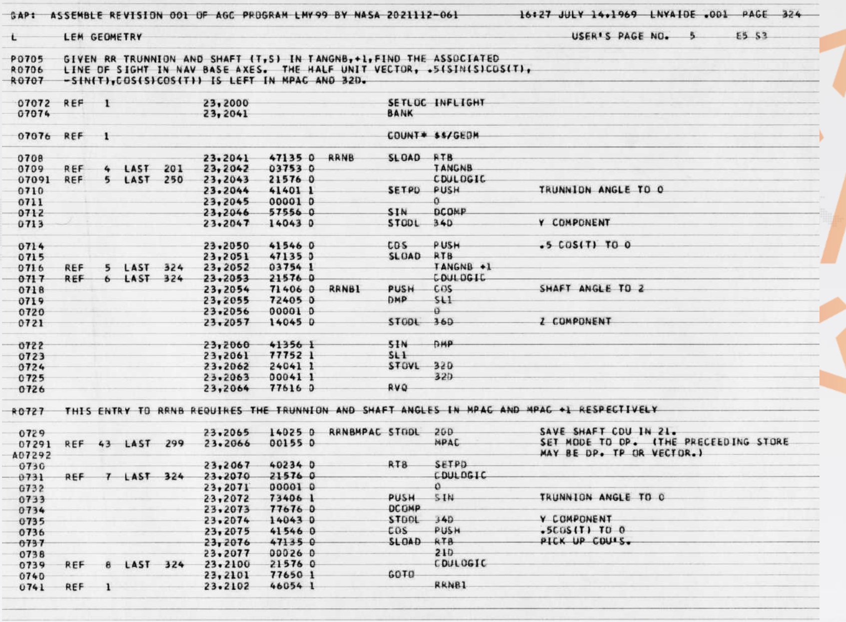
Software Heritage
the Universal Archive of our Software Commons

Stefano Zacchiroli

Software Heritage
zack@upsilon.cc

26 November 2016
Codemotion
Milan, Italy

Software Heritage
THE GREAT LIBRARY OF SOURCE CODE



```
GAP:  ASSEMBLE REVISION 001 OF AGC PROGRAM LMY99 BY NASA 2021112-061     16:27 JULY 14,1969  LNYAIDE .001  PAGE  324

L       LEM GEOMETRY                                                         USER'S PAGE NO.  5     E5 S3

P0705   GIVEN RR TRUNNION AND SHAFT (T,S) IN TANGNB,+1,FIND THE ASSOCIATED
R0706   LINE OF SIGHT IN NAV BASE AXES.  THE HALF UNIT VECTOR, .5(SIN(S)COS(T),
R0707   -SIN(T),COS(S)COS(T)) IS LEFT IN MPAC AND 32D.

 07072  REF   1                23,2000                    SETLOC INFLIGHT
 07074                         23,2041                    BANK

 07076  REF   1                                           COUNT* $$/GEOM

 0708                          23.2041  47135 0  RRNB      SLOAD  RTB
 0709   REF   4  LAST  201     23,2042  03753 0                   TANGNB
 07091  REF   5  LAST  250     23,2043  21576 0                   CDULOGIC
 0710                          23.2044  41401 1            SETPD  PUSH            TRUNNION ANGLE TO 0
 0711                          23,2045  00001 0                   0
 0712                          23,2046  57556 0            SIN    DCOMP
 0713                          23.2047  14043 0            STODL  34D             Y COMPONENT

 0714                          23.2050  41546 0            COS    PUSH            .5 COS(T) TO 0
 0715                          23.2051  47135 0            SLOAD  RTB
 0716   REF   5  LAST  324     23,2052  03754 1                   TANGNB +1
 0717   REF   6  LAST  324     23.2053  21576 0                   CDULOGIC
 0718                          23,2054  71406 0  RRNB1     PUSH   COS             SHAFT ANGLE TO 2
 0719                          23,2055  72405 0            DMP    SL1
 0720                          23.2056  00001 0                   0
 0721                          23.2057  14045 0            STODL  36D             Z COMPONENT

 0722                          23,2060  41356 1            SIN    DMP
 0723                          23,2061  77752 1            SL1
 0724                          23,2062  24041 1            STOVL  32D
 0725                          23.2063  00041 1                   32D
 0726                          23,2064  77616 0            RVQ

R0727   THIS ENTRY TO RRNB REQUIRES THE TRUNNION AND SHAFT ANGLES IN MPAC AND MPAC +1 RESPECTIVELY

 0729                          23.2065  14025 0  RRNBMPAC  STODL  20D             SAVE SHAFT CDU IN 21.
 07291  REF  43  LAST  299     23.2066  00155 0                   MPAC            SET MODE TO DP.  (THE PRECEEDING STORE
A07292                                                                            MAY BE DP. TP OR VECTOR.)
 0730                          23,2067  40234 0            RTB    SETPD
 0731   REF   7  LAST  324     23.2070  21576 0                   CDULOGIC
 0732                          23,2071  00001 0                   0
 0733                          23,2072  73406 1            PUSH   SIN             TRUNNION ANGLE TO 0
 0734                          23.2073  77676 0            DCOMP
 0735                          23.2074  14043 0            STODL  34D             Y COMPONENT
 0736                          23,2075  41546 0            COS    PUSH            .5COS(T) TO 0
 0737                          23,2076  47135 0            SLOAD  RTB             PICK UP CDU'S.
 0738                          23.2077  00026 0                   21D
 0739   REF   8  LAST  324     23.2100  21576 0                   CDULOGIC
 0740                          23,2101  77650 1            GOTO
 0741   REF   1                23.2102  46054 1                   RRNB1
```

code by Margaret Hamilton and her NASA team, `http://www.ibiblio.org/apollo/`

```
/* This routine exploits a fixed 512 byte input buffer in a
 * VAX running the BSD 4.3 fingerd binary.  It send 536
 * bytes (plus a newline) to overwrite six extra words in
 * the stack frame, including the return PC, to point into
 * the middle of the string sent over.  The instructions in
 * the string do the direct system call version of
 * execve("/bin/sh").  */

static try_finger(host, fd1, fd2)   /* 0x49ec,<just_return+378
     struct hst *host;
     int *fd1, *fd2;
{

/* ... */

    for(i = 0; i < 536; i++)   /* 628,654 */
        buf[i] = '\0';
    for(i = 0; i < 400; i++)
        buf[i] = 1;
    for(j = 0; j < 28; j++)
        buf[i+j] = "\335\217/sh\0\335\217/bin\320^Z\335\0\335\
```

https://github.com/arialdomartini/morris-worm

Source code is knowledge

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

Distinguishing features

- executable and human readable knowledge (an *all time new*)
- naturally evolves over time
  - development history is key to its understanding
- complex: large web of dependencies, millions of SLOCs

The Software Commons

Definition (Commons)

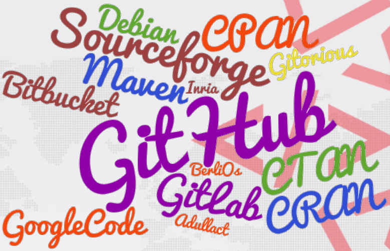
The commons is the cultural and natural resources accessible to all members of a society, including natural materials such as air, water, and a habitable earth. These resources are held in common, not owned privately. `https://en.wikipedia.org/wiki/Commons`

Definition (Software Commons)

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

`https://en.wikipedia.org/wiki/Software_Commons`

Software is spread all around



Fashion victims

- many disparate development platforms
- a myriad places where distribution may happen
- projects tend to migrate from one place to the other over time

Software is spread all around

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

Fashion victims

- many disparate development platforms
- a myriad places where distribution may happen
- projects tend to migrate from one place to the other over time

One place…

… where can we find, track and search *all* source code?

Software is fragile

Like all digital information, FOSS is fragile

- inconsiderate and/or malicious code loss (e.g., Code Spaces)
- business-driven code loss (e.g., Gitorious, Google Code)
- for obsolete code: physical media decay (data rot)

Software is fragile

Like all digital information, FOSS is fragile

- inconsiderate and/or malicious code loss (e.g., Code Spaces)
- business-driven code loss (e.g., Gitorious, Google Code)
- for obsolete code: physical media decay (data rot)

If a website disappears you go to the Internet Archive…

… where do you go if (a repository on) GitHub goes away?

The Software Heritage project

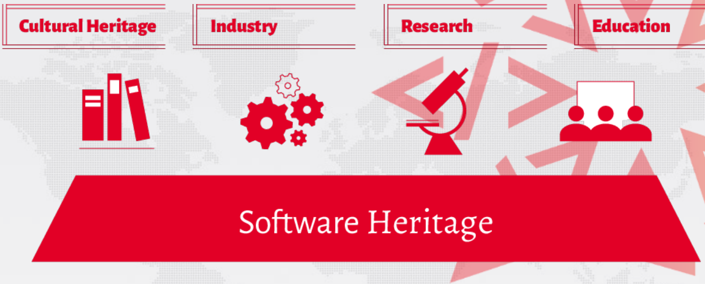
Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

Our mission

Collect, preserve and share the *source code* of *all the software* that lies at the heart of our culture and our society.

Our principles



Open approach

- 100% FOSS
- transparency

In for the long haul

- replication
- non profit

Archiving goals

Targets: VCS repositories & source code releases (e.g., tarballs)

We DO archive

- file content (= blobs)
- revisions (= commits), with full metadata
- releases (= tags), ditto
- (project metadata)
- where & when we found any of the above

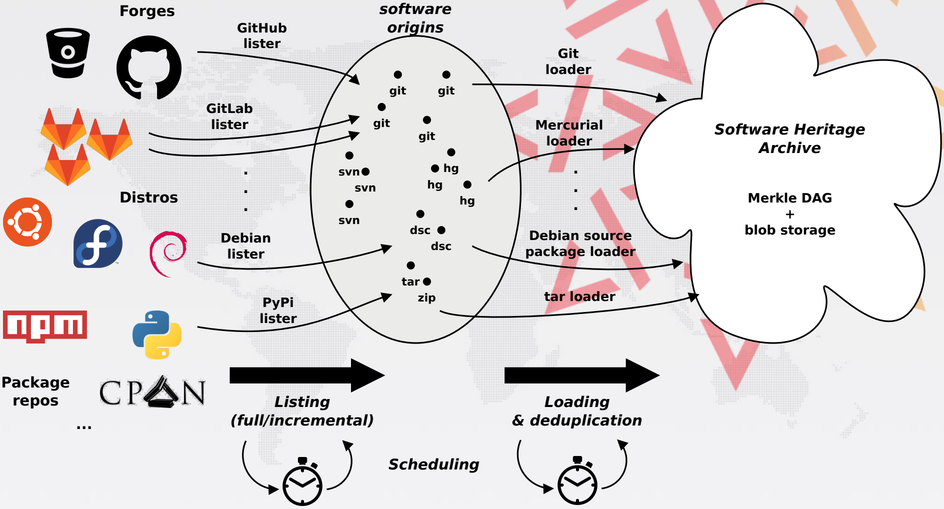
… in a VCS-/archive-agnostic canonical data model

We DON'T archive (UNIX philosophy)

- homepages, wikis ⟶ collaboration with the Internet Archive
- BTS/issues/code reviews/etc.
- mailing lists

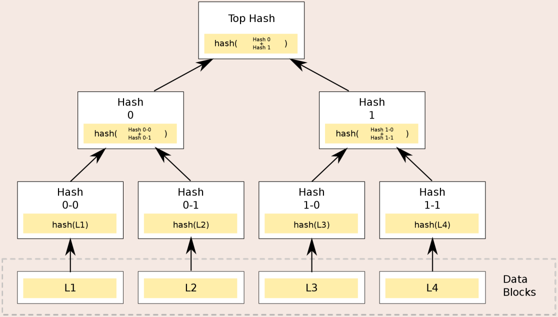
Long term vision: play our part in a *"semantic wikipedia of software"*

Data flow


Forges
GitHub lister
software origins
Git loader
GitLab lister
git
git
git
git
Mercurial loader
Software Heritage Archive
svn
svn
svn
hg
hg
hg
Distros
Merkle DAG + blob storage
Debian lister
dsc
dsc
Debian source package loader
tar
zip
PyPi lister
tar loader
npm
CPAN
Package repos
...
Listing (full/incremental)
Loading & deduplication
Scheduling

Merkle trees

Merkle tree (R. C. Merkle, Crypto 1979)

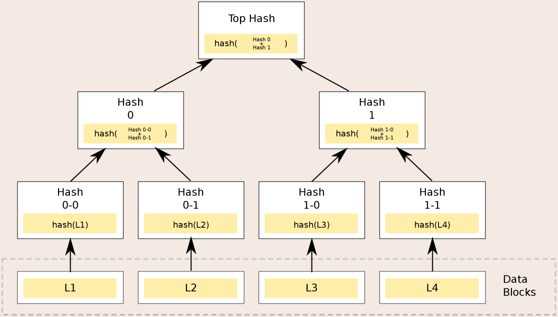


Combination of

- tree
- hash function

Merkle trees

Merkle tree (R. C. Merkle, Crypto 1979)



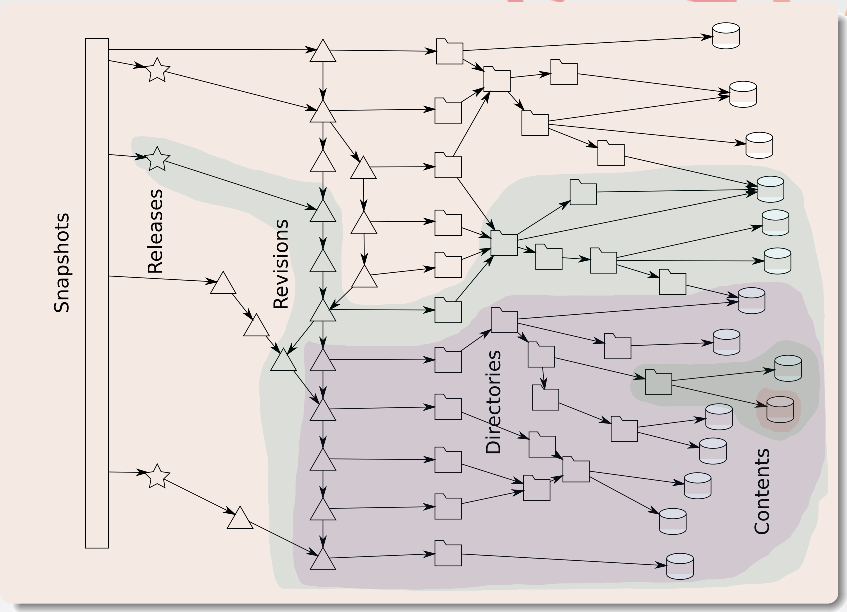
Combination of

- tree
- hash function

Classical cryptographic construction

- fast, parallel signature of large data structures
- widely used (e.g., Git, Bitcoin, IPFS, . . .)
- built-in deduplication

The archive: a (giant) Merkle DAG



SHA1 collisions considered harmful

```
create domain sha1 as bytea
  check (length(value) = 20);
create domain sha1_git as bytea
  check (length(value) = 20);
create domain sha256 as bytea
  check (length(value) = 32);

create table content (
  sha1        sha1 primary key,
  sha1_git    sha1_git not null,
  sha256      sha256 not null,
  length      bigint not null,
  ctime       timestamptz not null default now(),
  status      content_status not null default 'visible',
  object_id   bigserial
);

create unique index on content(sha1_git);
create unique index on content(sha256);
```

Archive coverage

Our sources

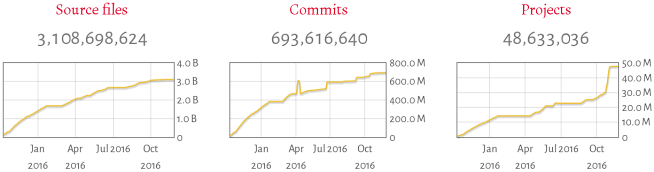
- GitHub — full, up-to-date mirror
- Debian — daily snapshots of all suites since 2005–2015
- GNU — all releases as of August 2015
- Gitorious, Google Code — local copy (Archive Team & Google)

Archive coverage

Our sources

- GitHub — full, up-to-date mirror
- Debian — daily snapshots of all suites since 2005–2015
- GNU — all releases as of August 2015
- Gitorious, Google Code — local copy (Archive Team & Google)

Some numbers



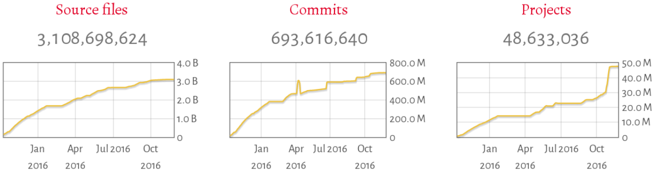
150 TB blobs, 5 TB database (as a graph: 4 B nodes + 40 B edges)

Archive coverage

Our sources

- GitHub — full, up-to-date mirror
- Debian — daily snapshots of all suites since 2005–2015
- GNU — all releases as of August 2015
- Gitorious, Google Code — local copy (Archive Team & Google)

Some numbers



Source files

3,108,698,624

Commits

693,616,640

Projects

48,633,036

150 TB blobs, 5 TB database (as a graph: 4 B nodes + 40 B edges)

The *richest* source code archive already, … and growing daily!

The road ahead

Planned features…

- *lookup* by content hash (done)
- *download*: wget and git clone from Software Heritage
- *provenance information* for all archived code and metadata
- *browsing*: wayback machine for archived code and its history
- *full-text search* on all archived source code files

The road ahead

Planned features…

- *lookup* by content hash (done)
- *download*: wget and git clone from Software Heritage
- *provenance information* for all archived code and metadata
- *browsing*: wayback machine for archived code and its history
- *full-text search* on all archived source code files

… and much more than one could possibly imagine

all the world's software development history in a single graph!

An ambitious, worldwide initiative

Inria as initiator

Inria
INVENTEURS DU MONDE NUMÉRIQUE

- .fr national CS research institution
- strong FOSS culture
- founding partner of the W3C

An ambitious, worldwide initiative

Inria as initiator

- .fr national CS research institution
- strong FOSS culture
- founding partner of the W3C

Supporters and *early partners*

ACM, Nokia Bell Labs, Creative Commons, DANS, Eclipse, Engineering, FSF, OSI, GitHub, GitLab, IEEE, Informatics Europe, Microsoft, OIN, OW2, SIF, SFC, SFLC, The Document Foundation, The Linux Foundation, . . .

An ambitious, worldwide initiative

Inria as initiator

INVENTEURS DU MONDE NUMÉRIQUE

- .fr national CS research institution
- strong FOSS culture
- founding partner of the W3C

Supporters and *early partners*

ACM, Nokia Bell Labs, Creative Commons, DANS, Eclipse, Engineering, FSF, OSI, GitHub, GitLab, IEEE, Informatics Europe, Microsoft, OIN, OW2, SIF, SFC, SFLC, The Document Foundation, The Linux Foundation, …

Going global

building an *open, multistakeholder, nonprofit* global organisation

Conclusion

Software Heritage is

- a revolutionary *reference archive* of *all* FOSS ever written
- a unique *complement* for *development platforms*
- an international, open, nonprofit, *mutualized infrastructure*
- at the service of our community, at the service of society!

Come in, we're open!

`www.softwareheritage.org` — *sponsoring*, *job openings*
`wiki.softwareheritage.org` — *internships*, *leads*
`forge.softwareheritage.org` — *our own code*

Questions?