

Software Heritage

Collecting, preserving and sharing all our Source Code

Roberto Di Cosmo

`roberto@dicosmo.org`

September 5th, 2018



Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software
- 7 Building for the long term
- 8 Conclusion

Software is everywhere



Source code is *executable* and *human readable* knowledge

a growing part of our *Cultural Heritage*

Source code is *special*

Harold Abelson, Structure and Interpretation of Computer Programs

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

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;
}
```

Net. queue in Linux (excerpt)

```
/*
 * SFB uses two B[1][n] : L x N arrays of bins (L levels, N bins per level)
 * This implementation uses L = 8 and N = 16
 * This permits us to split one 32bit hash (provided per packet by rxhash or
 * external classifier) into 8 subhashes of 4 bits.
 */
#define SFB_BUCKET_SHIFT 4
#define SFB_NUMBUCKETS (1 << SFB_BUCKET_SHIFT) /* N bins per Level */
#define SFB_BUCKET_MASK (SFB_NUMBUCKETS - 1)
#define SFB_LEVELS (32 / SFB_BUCKET_SHIFT) /* L */

/* SFB algo uses a virtual queue, named "bin" */
struct sfb_bucket {
    u16 qlen; /* length of virtual queue */
    u16 p_mark; /* marking probability */
};
```

Len Shustek, Computer History Museum

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

~ 50 years, a lightning fast growth

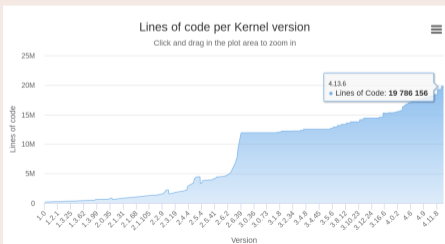
Apollo 11 Guidance Computer (~60.000 lines), 1969



"When I first got into it, nobody knew what it was that we were doing. It was like the Wild West."

Margaret Hamilton

Linux Kernel



... now in your pockets!

are we taking care of all this?

Software is spread all around

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



A word cloud centered on a world map background. The words are in various colors and sizes, representing concepts related to software fragility. The largest words are 'damage', 'disaster', 'malicious', 'obsolete', 'attack', and 'deletion'. Other words include 'media', 'aging', 'tear', 'dependencies', 'dangling', 'wear', 'corruption', 'encryption', 'format', 'reference', and 'storage'. The background features a faint world map and a decorative pattern of red and orange triangles on the right side.

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

Software lacks its own research infrastructure



Photo: ALMA(ESO/NAOJ/NRAO), R. Hills

Today we are at a turning point

Looking at the past

- a lot of old software misplaced, lost, or behind barriers, but...
- most founding fathers are still here, and willing to share
- **urgent** to collect their knowledge

Only a few years left.

Looking at the future

- software development and use skyrockets: more programmers, and more code!
- **essential** to provide a **universal** platform for all the future software source code

Every year that goes by makes the problem worse.

it is **urgent** to take action!

Outline

- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software
- 7 Building for the long term
- 8 Conclusion



Software Heritage



Our mission

Collect, preserve and share the source code of all the software that is available

Past, present and future

Preserving the past, enhancing the present, preparing the future

Cultural Heritage



Industry



Research



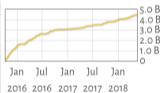
Education



Software Heritage

Source files

4,536,067,027



Commits

1,024,675,748



Projects

83,801,775



Technology

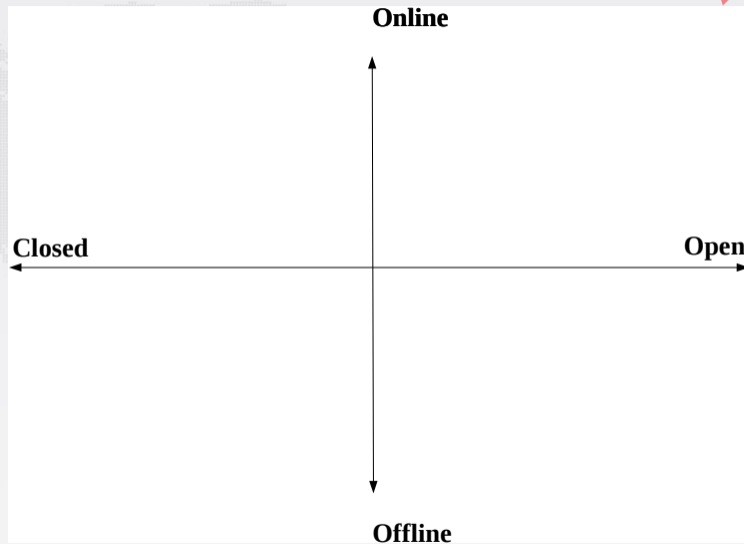
- transparency and FOSS
- replicas all the way down

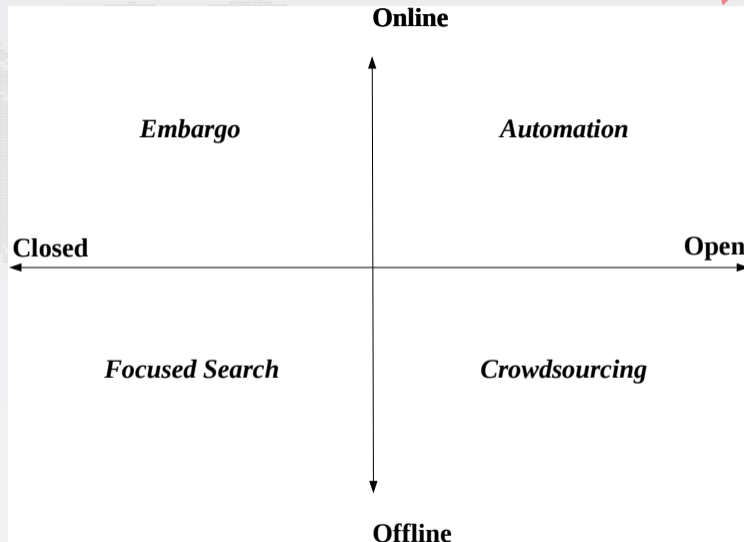
Content

- intrinsic identifiers
- facts and provenance

Organization

- non-profit
- **mirror network**

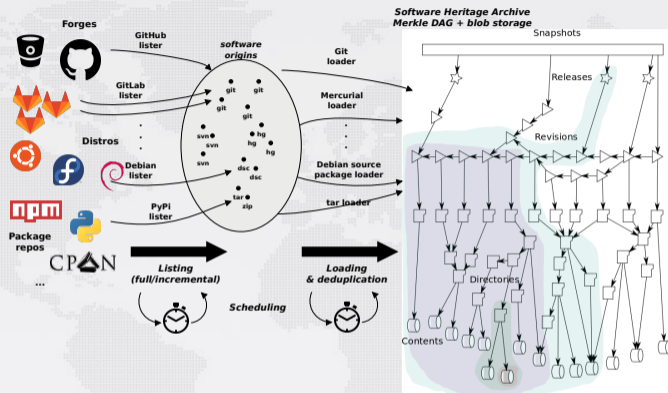




- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture**
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software
- 7 Building for the long term
- 8 Conclusion



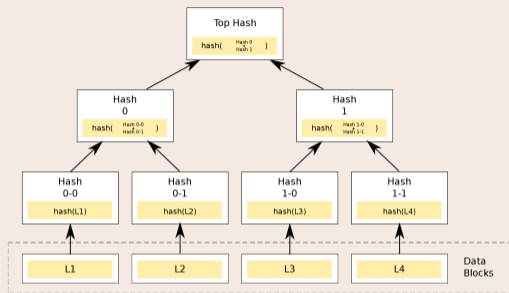
Automation (first quadrant), and storage



- full development history permanently archived
- origins: GitHub (auto), Debian (auto), [Gitlab.com](https://www.gitlab.com), Gitorious, Google Code, GNU
- ~ 200Tb raw contents, ~ 10Tb graph (10Bn nodes, 100Bn edges)

Much more than an archive!

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, blockchains, IPFS, ...)
- built-in deduplication

A bird's eye view

origin https://forge.softwareheritage.org/source/helloworld.git
visit 1
timestamp Fri Feb 9 12:38:45 2018 +0100
snapshot 0861db5e...

origin https://forge.softwareheritage.org/...
visit 2
timestamp Fri Feb 9 13:29:00 2018 +0100
snapshot 0861db5e...

origin https://forge.softwareheritage.org/...
visit 3
timestamp Fri Feb 9 15:52:50 2018 +0100
snapshot 510aa88b...

```
<<Snapshot>>  
510aa88b...  
+branches  
HEAD  
refs/heads/master  
refs/heads/doc  
refs/tags/1.0
```

```
<<Snapshot>>  
0861db5e...  
+branches  
HEAD  
refs/heads/master  
refs/tags/1.0
```

```
<<Releases>>  
edf82f21...  
+author = "Foo Bar <foo@...>"  
+name = "1.0"  
+message = "1.0 release"  
+timestamp = Thu Feb 8 15:51:00 2018 +0100  
+target
```

```
<<Revision>>  
1a99a56b...  
+author = "Foo Bar <foo@...>"  
+message = "Merge branch 'doc'"  
+timestamp = Fri Feb 9 15:44:45 2018 +0100  
+directory: Directory  
+parents: Revision list
```

```
<<Revision>>  
3d515253...  
+author = "Foo Bar <foo@...>"  
+message = "README: add homepage link"  
+timestamp = Fri Feb 9 15:44:30 2018 +0100  
+directory: Directory  
+parents: Revision list
```

```
<<Revision>>  
c7640e8d...  
+author = "Foo Bar <foo@...>"  
+message = "move source code to src/\n..."  
+timestamp = Thu Feb 8 15:26:08 2018 +0100  
+directory: Directory  
+parents: Revision list
```

```
<<Revision>>  
43ef7dcd...  
+author = "Foo Bar <foo@...>"  
+message = "add licensing information and README"  
+timestamp = Thu Feb 8 10:54:09 2018 +0100  
+directory: Directory  
+parents: Revision list
```

```
<<Revision>>  
a3ee21ad...  
+author = "Foo Bar <foo@...>"  
+message = "add build toolchain ..."  
+timestamp = Thu Feb 8 10:49:29 2018 +0100  
+directory: Directory  
+parents: Revision list
```

```
<<Revision>>  
1886826f...  
+author = "Foo Bar <foo@...>"  
+message = "implement a trivial ..."  
+timestamp = Thu Feb 8 10:44:35 2018 +0100  
+directory: Directory  
+parents: Revision list = None
```

```
<<Directory>>  
ded70c63...  
+entries  
"COPYING"  
"Makefile"  
"README.md"  
"src"
```

```
<<Directory>>  
ded70c63...  
+entries  
",.gitignore"  
"COPYING"  
"Makefile"  
"README.md"  
"hello.c"
```

```
<<Directory>>  
45f0c078...  
+entries  
"COPYING"  
"Makefile"  
"README.md"  
"src"
```

```
<<Directory>>  
fa8c0908...  
+entries  
",.gitignore"  
"COPYING"  
"Makefile"  
"README.md"  
"hello.c"
```

```
<<Directory>>  
b94a90cd...  
+entries  
",.gitignore"  
"COPYING"  
"Makefile"  
"hello.c"
```

```
<<Directory>>  
6ca2e444...  
+entries  
"hello.c"
```

```
<<Content>>  
4ec6b1e9...  
+data = "...For more info..."
```

```
<<Content>>  
a8becc46...  
+data = "SRC_DIR = ..."
```

```
<<Content>>  
a1afd006...  
+data = "...Yet another..."
```

```
<<Content>>  
94a9e02...  
+data = "...GNU GENERAL..."
```

```
<<Content>>  
59b32b2f...  
+data = "...\nhello"
```

```
<<Content>>  
225ae01b...  
+data = "all: hello\n\n..."
```

```
<<Content>>  
c839dea9...  
+data = "#include ..."
```

Archive content
after visits 1, 2 and 3

Archive content
after visits 1 and 2

Outline

- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software
- 7 Building for the long term
- 8 Conclusion

Reference archive for all software

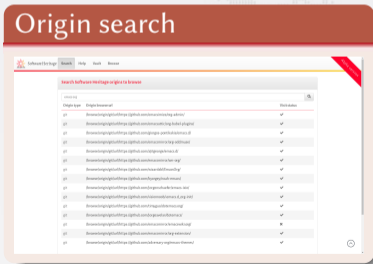
A "wayback machine" for software source code ...

with **intrinsic identifiers!**

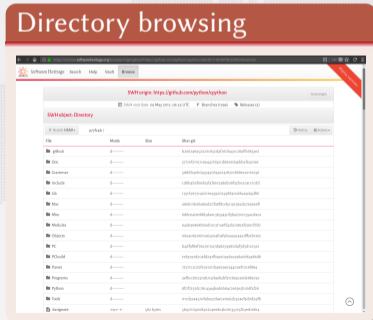
- <http://archive.softwareheritage.org/browse>
- <http://bit.ly/swhpids> for persistent identifiers

Demo time: let's highlight some features...

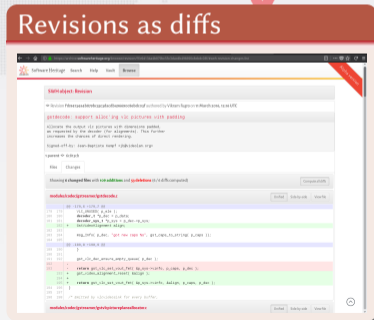
Origin search



Directory browsing



Revisions as diffs



Outline

- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software
- 7 Building for the long term
- 8 Conclusion

Research software: a long way to go!

ICSE (Zannier, Melrik, Maurer, 2006)

- complete absence of replication studies

ACM TOSEM 2001 to 2006

C. Ghezzi <http://bit.ly/tosemreprod>

- 60% of all papers have tools: **only 20% installable**

Collberg's 2015 study

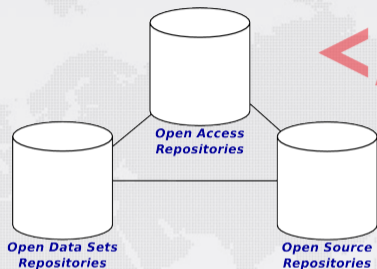
<http://reproducibility.cs.arizona.edu/>

- 601 mainstream papers: 508 with tools, **only 40% installable**

Main reasons

source code (*or the right version of it*) cannot be found

Supporting more accessible and reproducible science

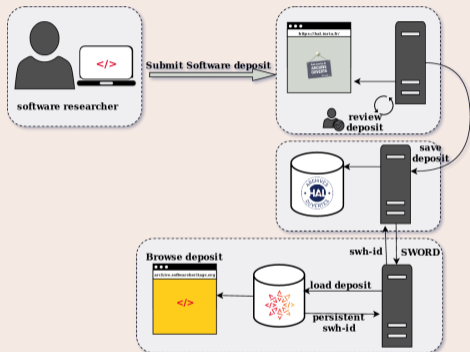


A global library referencing all software used in all research fields

- completes the infrastructure for **Open Access** in science
- provides **intrinsic persistent identifiers** for scientific **reproducibility**
- enables large scale, verifiable **software studies**

Paper points to lost source code on gitorious

- https://www.openaire.eu/search/publication?articleId=dedup_wf_001::cd996f0b6236b90659f84f99feb62bcc
- <https://gitorious.org/parmap>
- <https://archive.softwareheritage.org/browse/search/?url=%22gitorious.org/parmap%22>



Generic mechanism:

- SWORD based
- review process
- versioning

How to do it:

- **today:** deposit .zip or .tar.gz file (*guide*)
- **tomorrow:**
 - provide *SWH id* and metadata
 - include *metadata file* for automatic metadata extraction
 - ...

September 2018: **open to all** on <https://hal.archives-ouvertes.fr/>

The way to go to archive and reference scientific software

All features of Software Heritage *for free*

- **intrinsic IDs** (integrity, not dependent on resolvers!), browse, download (now)
- metadata, licenses, provenance analysis (plagiarism detection), classification (wip)
- and many more (powerful connections with SE and Industry)

Coverage and uniformity

- **one** archive for **all** domains (industry included)
- you can reference *any* software, not just the deposited one
(thanks D. Katz for pointing this out)
- **git-compatible** identifiers greatly simplify workflows

Sustainability

... doors are open!

one infrastructure

independent non profit foundation

worldwide mirrors

Outline

- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software**
- 7 Building for the long term
- 8 Conclusion



Large scale *repeatable* software studies...

- vulnerability detection
- dependency analysis
- pattern elicitation
- automatic classification ...

... need a uniform representation

Software Heritage has **one data model** for all forges/VCS...

... yes, we do **data normalization** of software evolution!

Breaking news: *soon* an **Amazon public data set!**

Outline

- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software
- 7 Building for the long term**
- 8 Conclusion

Growing Support

Landmark Inria Unesco agreement, April 3rd, 2017



Sharing the vision



Contributing to the mission



The next steps

The Software Heritage Foundation

- independent
- long term mission
- multistakeholder

The community

- academia: Open Access, research
- industry: better software
- cultural heritage: **all** the software history

The mirror network

- resilience
- biodiversity

“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

You can help!

Tackle the research challenges

machine learning, classification, efficient graph queries, mirror protocols, ...

Contribute

- forge.softwareheritage.org
- www.softwareheritage.org/jobs

EU Copyright directive: ACT NOW to protect software development!

savecodeshare.eu

saveyourinternet.eu

Funding

- bring in new partners/sponsors :
sponsorship.softwareheritage.org
- give *your own contribution* :
www.softwareheritage.org/donate

Spread the word!

- *use* the archive and help others do
- tell everybody about Software Heritage

- 1 Software is everywhere and nowhere
- 2 The Software Heritage initiative
- 3 Architecture
- 4 Using the Software Heritage archive
- 5 Open Science
- 6 Science of Software
- 7 Building for the long term
- 8 Conclusion



Software Heritage

www.softwareheritage.org

@swheritage

Library of Alexandria of code

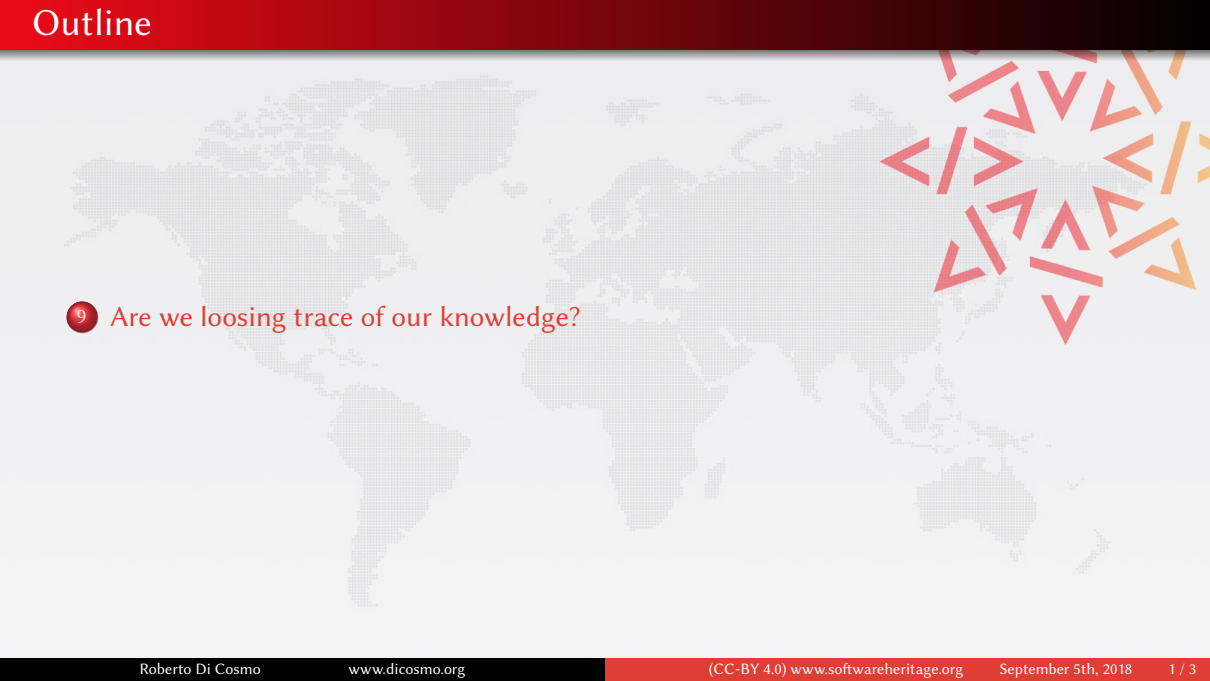


- recover the past
- structure the future

A CERN for Software



- build better software
 - for industry
 - for society as a whole



9 Are we losing trace of our knowledge?

URL decay disrupts the *web of reference*

Web links *are not* permanent (even *permalinks*)

there is no general guarantee that a URL... which at one time points to a given object continues to do so

T. Berners-Lee et al. Uniform Resource Locators. RFC 1738.

404

URLs used in articles *decay!*

Analysis of *IEEE Computer* (Computer), and the *Communications of the ACM* (CACM): 1995-1999

- the *half-life* of a referenced URL is *approximately 4 years* from its publication date
D. Spinellis. The Decay and Failures of URL References.

Communications of the ACM, 46(1):71-77, January 2003.

Similar findings in Lawrence, S. et al. *Persistence of Web References in Scientific Research*, *IEEE Computer*, 34(2), pp. 26-31, 2001.

An example from Astronomy

Domain	links (broken)	.html	.txt	.dat	.gz	.tar	.fits	tilde
cxk.harvard.edu	802 (110)	336 (70)	0	0	4 (2)	5 (4)	1	0
heasarc.gsfc.nasa.gov	640 (33)	423 (27)	1	0	0	0	0	0
www.stsci.edu	498 (61)	205 (29)	3	0	0	0	0	15 (10)
esc.harvard.edu	471 (152)	212 (99)	0	0	0	0	0	1 (1)
ssc.spitzer.caltech.edu	427 (194)	125 (76)	3 (3)	0	0	0	0	0
cfa-www.harvard.edu	352 (68)	277 (52)	1	0	0	0	0	54 (17)
archive.stsci.edu	308 (58)	57 (9)	2	1 (0)	0	0	0	0
www.ipac.caltech.edu	285 (14)	209 (12)	0	0	0	0	0	0
www.atnf.csiro.au	211 (21)	12 (6)	0	0	0	0	0	7 (5)
space.mit.edu	193 (10)	58 (5)	1	0	0	0	0	2 (1)
www.astro.psu.edu	186 (4)	103 (1)	1	10	1	1	0	2
www.eso.org	186 (58)	54 (22)	1 (1)	0	0	0	0	4 (1)
isa.ipac.caltech.edu	163 (5)	38	0	0	1	0	0	0
www.sdsu.org	156 (2)	106 (1)	0	0	0	0	0	0
hea-www.harvard.edu	125 (37)	42 (17)	1	0	0	1	0	26 (16)
physics.nist.gov	125 (3)	63 (2)	0	0	0	0	0	0
www.noao.edu	120 (3)	50 (2)	0	0	0	0	0	0
xmm.vilspa.esa.es	118 (35)	23 (19)	0	0	8 (1)	0	0	1 (1)
www.astro.princeton.edu	115 (31)	43 (14)	0	0	0	0	0	53 (12)
ed.usno.navy.mil	110 (27)	98 (22)	3 (3)	0	0	0	0	1 (1)

This table lists total number of links and broken links (HTTP status codes 3xx, 4xx, and 5xx) to top domains (domains with over 100 links) found within articles published in the four main astronomy journals between 1997 and 2008. The table also shows, for each domain, the portion of links to common filename extensions, as well as links that contain the tilde character.
doi:10.1371/journal.pone.0104798.t001

How Do Astronomers Share Data?

Pepe, Goodman, Muench, Crosas, Erdmann

[dx.doi.org/10.1371/journal.pone.0104798](https://doi.org/10.1371/journal.pone.0104798)

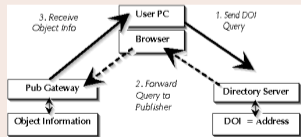
PLOS August 28, 2014

Example: doi:10.1109/MSR.2015.10

- to find what 10.1109/MSR.2015.10 is, go to a *resolver* (e.g. doi.org)
- this returns <http://ieeexplore.ieee.org/document/7180064/>
- at this URL we find ...

The screenshot shows a web page with the title "Mining Component Repositories for Instability Issues". It features a navigation bar with "View Document" and "Full Text" buttons. Below the navigation bar, there is a table with columns for "Abstract", "Authors", "Figures", "References", "Citations", "Keywords", "Metrics", and "Links". The "Abstract" section contains text about component repositories and their role in software life cycle management. The "Published in" section lists "Mining Software Repositories (MSR), 2015 IEEE/ACM 12th Mining Conference on". The "Date of Conference" is "01-17 May 2015". The "Date Added to IEEE Xplore" is "01 August 2015". The "Electronic ISBN" is "978-0-7691-0244-2". The "Publisher" is "IEEE".

Architecture of the DOI infrastructure



- DOI resolution *can change*
- content at URL *can change*
- no *intrinsic* way of noticing
- persistence based on *good will* of *multiple parties*