

Software Heritage

Pourquoi et comment construire l'archive universel du logiciel

Roberto Di Cosmo

`roberto@dicosmo.org`

2 Mai, 2018

5e Colloque libre de l'Adte 2018



Software Heritage

THE GREAT LIBRARY OF SOURCE CODE

- 
- 1 Présentation
 - 2 Le logiciel tout autour de nous
 - 3 Le logiciel: le grand absent dans la Science!
 - 4 Software Heritage
 - 5 Construire avec une vision à long terme

Roberto Di Cosmo en quelques lignes

Professeur d'Informatique à Paris, en détachement INRIA

- 30 ans de recherche et enseignement (Info. Theor., Programmation, Génie Logiciel, Erdos #: 3)
- 20 ans de logiciel libre
- 10 ans créant et dirigeant des structures pour le bien commun



1998 *Holdup Planétaire* – voix du LL dans la francophonie

1999 *DemoLinux* – première distribution GNU/Linux live

2007 *Groupe Thématique Logiciel Libre*
150 members 40 projects 200Me

2010 *IRILL* www.irill.org

2015 *OCaml MOOC* doi.org/10.1145/3110248

2015 *Software Heritage* avec INRIA

- 
- 1 Présentation
 - 2 Le logiciel tout autour de nous
 - 3 Le logiciel: le grand absent dans la Science!
 - 4 Software Heritage
 - 5 Construire avec une vision à long terme



Le logiciel contient notre **Connaissance** et notre **Patrimoine Culturel**

Le code source est important!



"The source code for a work means the preferred form of the work for making modifications to it."
— GPL Licence

Hello World

Logiciel (extrait de l'exécutable)

```
4004e6: 55
4004e7: 48 89 e5
4004ea: bf 84 05 40 00
4004ef: b8 00 00 00 00
4004f4: e8 c7 fe ff ff
4004f9: 90
4004fa: 5d
4004fb: c3
```

Logiciel (code source)

```
/* Hello World program */

#include<stdio.h>

void main()
{
    printf("Hello World");
}
```

Le code source d'un logiciel est *special*

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.”

Quake 2 (extrait)

```
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, Linux (extrait)

```
/*
 * SFB uses two B[l][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.”

Le Logiciel Libre est un Bien Commun

Définition (Bien Commun)

Un **commun** est un système ouvert avec, au centre, une ou plusieurs ressources partagées, gérées collectivement par une communauté ; ... Ces ressources peuvent être naturelles : une forêt, une rivière ; matérielles : une machine-outil, une maison, une centrale électrique ; immatérielles : une connaissance, **un logiciel**.

<https://fr.wikipedia.org/wiki/Communs>

Le code source des logiciels libres

est une partie précieuse de nos biens communs

en ~ 50 ans, une croissance spectaculaire

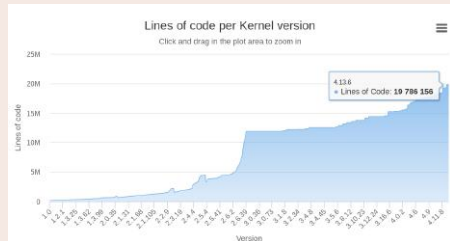
Apollo 11 Guidance Computer (~60.000 lignes), 1969



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

Margaret Hamilton

Le noyau Linux



... maintenant dans nos poches!

est-ce qu'on prend bien soin de tout cela?

Le logiciel est éparpillé partout

A word cloud featuring various software-related terms and logos. The most prominent words are 'Sourceforge', 'Git', 'GitHub', 'Debian', 'CPAN', 'Maven', 'Bitbucket', 'GoogleCode', 'GitLab', 'CTan', and 'CRAN'. Other smaller words include 'Inria', 'Gitorious', 'BerliOs', and 'Adullact'. The word cloud is set against a background of a world map and a decorative starburst pattern.



A word cloud centered on the slide, featuring various terms related to software fragility and security. The words are arranged in a circular pattern, with some overlapping. The colors of the words include shades of purple, blue, green, and orange. The background of the slide features a faint world map and a decorative starburst pattern in the top right corner.

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

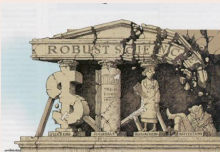
Nous n'avons pas d'infrastructure de recherche dédiée



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

- 
- 
- 1 Présentation
 - 2 Le logiciel tout autour de nous
 - 3 Le logiciel: le grand absent dans la Science!
 - 4 Software Heritage
 - 5 Construire avec une vision à long terme

"Une science Sub-prime"? (Nicholas Humphrey)



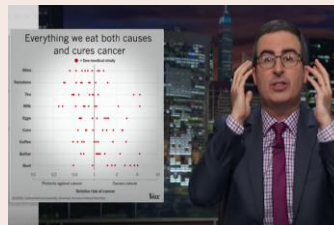
- inconsistences
- corruption de données, fraude
- resultats non reproductibles...

(image: Nature, Sep. 2015)

Cela commence à se savoir



October 2013



John Oliver, Science May 2016

Comment nous construisons la connaissance scientifique

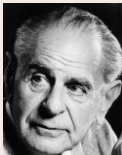
La méthode expérimentale



- on *observe*
- on formule des *hypotheses*
- on réalise un **experiment**
- on formule une *théorie*

Et ensuite on **reproduit** et on **vérifie**.

La réproducibilité est la clé



non-reproducible single occurrences are of no significance to science

Karl Popper, The Logic of Scientific Discovery, 1934

Quand il y a du logiciel dans une expérience, il nous faut
accès ouvert aux articles qui le décrivent
les données ouvertes utilisés dans l'expérience
le code source de tous les composants
l'environnement d'exécution
des références stables entre tout cela

Remarque

Les deux premiers points sont bien connus

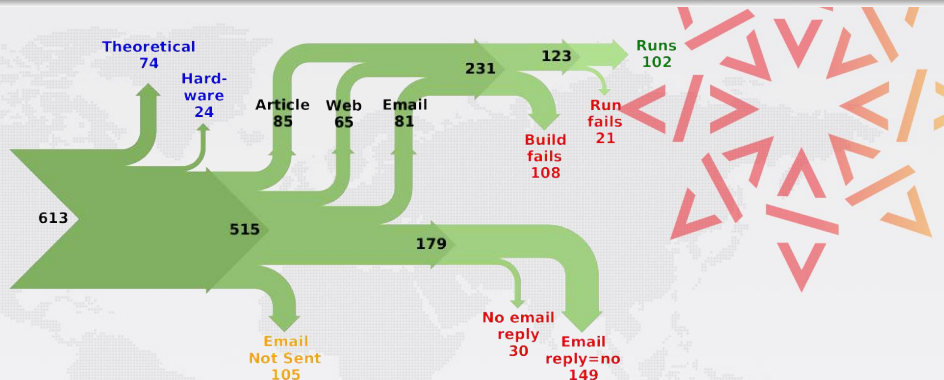
... mais pour le *logiciel*?

- 8 ACM conferences: ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12
- 5 journals: TACO'9, TISSEC'15, TOCS'30, TODS'37, TOPLAS'34

La question

[illegible]

Le résultat



On peut en débattre (<http://cs.brown.edu/~sk/Memos/Examining-Reproducibility/>), mais ...

... c'est 81% de recherche **non reproductible** !

En regardant vers le passé

- pas mal de logiciel perdu, égaré, ou prisonnier, mais...
- les personnes qui l'ont créé sont presque tous là, et veulent partager
- c'est **urgent** de recueillir leur connaissance

Nous n'avons que quelques années pour cela!

En regardant vers le futur

- chaque jour plus de programmeurs, chaque jour plus de code!
- **essentiel** de construire une plateforme **universelle** pour conserver tout le code futur

Chaque année qui passe, le problème grandit...

il est *urgent* d'agir!

- 
- 1 Présentation
 - 2 Le logiciel tout autour de nous
 - 3 Le logiciel: le grand absent dans la Science!
 - 4 Software Heritage
 - 5 Construire avec une vision à long terme



Software Heritage



Notre mission

Recolter, préserver et partager le *code source* de *tout le logiciel* disponible

Passé, présent et futur

Préserver le passé, améliorer le présent, préparer le futur



Technologie

- transparence et Logiciel Libre
- replication sur toute la ligne

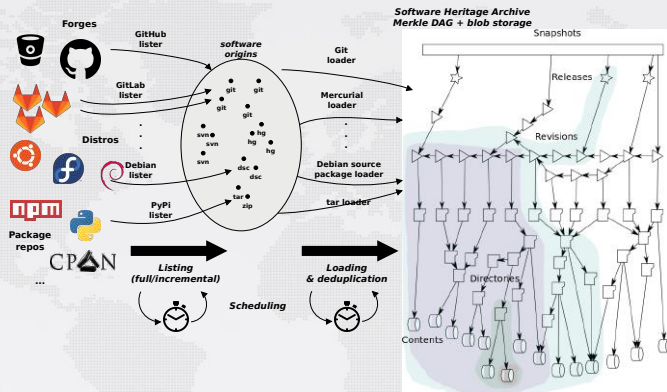
Contenus

- identifiants intrinsèques
- faits, et traçabilité

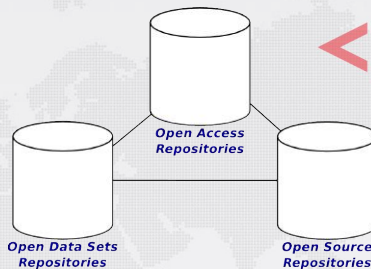
Organisation

- non-profit
- multi-partenaire

Architecture (simplifiée)



- toute l'histoire de développement archivée pour toujours
- origines: GitHub (auto), Debian (auto), Gitorious, Google Code, GNU
- ~150Tb contenus, ~10Tb graphe (7+Bn noeuds, 60+Bn arêtes)



Une archive universelle pour tous le logiciel scientifique

- complète l'infrastructure pour l'**Accès Ouvert** dans la Science
- fournit des identifiant persistants intrinsèques pour la **reproductibilité** scientifique
- perme des **études logicielles** à large échelle vérifiables

Mettons les mains à la pâte!

Preview *juste pour vous*:

une "wayback machine" du code source!

- allez à <http://archive.softwareheritage.org/browse>
- utilisez : `adte / 2018`

Depôt du logiciel scientifique via l'archive ouverte HAL

- voir le guide de dépôt sur <http://bit.ly/swhdeposithalfr>

- 
- 
- 1 Présentation
 - 2 Le logiciel tout autour de nous
 - 3 Le logiciel: le grand absent dans la Science!
 - 4 Software Heritage
 - 5 Construire avec une vision à long terme

Les soutiens arrivent

Accord historique Inria Unesco, 3 Avril 2017



Partagent la vision



Contribuent à l'effort



La Fondation Software Heritage

- indépendante
- mission de long terme
- multi-partenaire

Les communautés

- academique: Accès Ouvert, recherche
- industrie: du logiciel meilleur
- patrimoine culturel: l'histoire du logiciel

Le réseau de miroirs

- resilience
- biodiversité

“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

Bibliothèque d'Alexandrie du logiciel



Agir *maintenant* pour

- sauver l'histoire
 - les créateurs sont encore là
- structurer le futur
 - le développement explose

Un CERN pour le Logiciel



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

Une *infrastructure commune* pour

- les besoins industriels
- la recherche sur le logiciel
- une science meilleure
- au service de toute la société

Venez nous rejoindre, vous pouvez aider!



Software Heritage

www.softwareheritage.org

[@swheritage](https://twitter.com/swheritage)

On a besoin de vous!

partenariats, miroirs

<mailto:roberto@dicosmo.org>

sponsors

sponsorship.softwareheritage.org

donations

www.softwareheritage.org/donate

notre code

forge.softwareheritage.org