



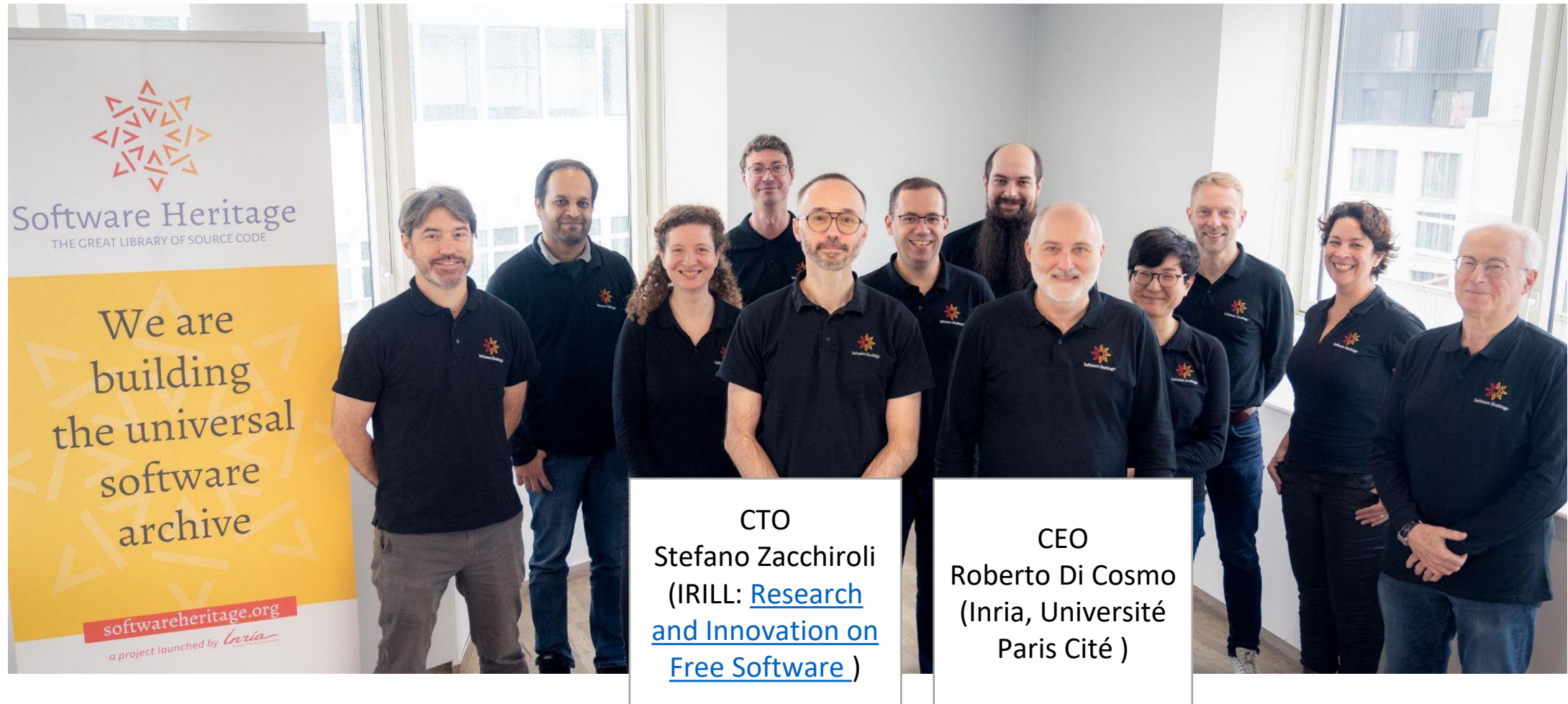
Software Heritage
THE GREAT LIBRARY OF SOURCE CODE

Archiver et référencer son code source pour une recherche plus traçable

2024-07-02 | ANF « Protéger, valoriser et faire vivre un logiciel libre »

sabrina.granger@inria.fr

L'équipe Software Heritage en janv. 2024 = 1 CEO + 1 CTO + 15 membres d'équipe



A quick reproducibility disclaimer

 Jon Claerbout

“[...] the ability to achieve similar or nearly identical results using comparable materials and methodologies.

This term may be used when specific findings from a study are obtained by an independent group of researchers.”

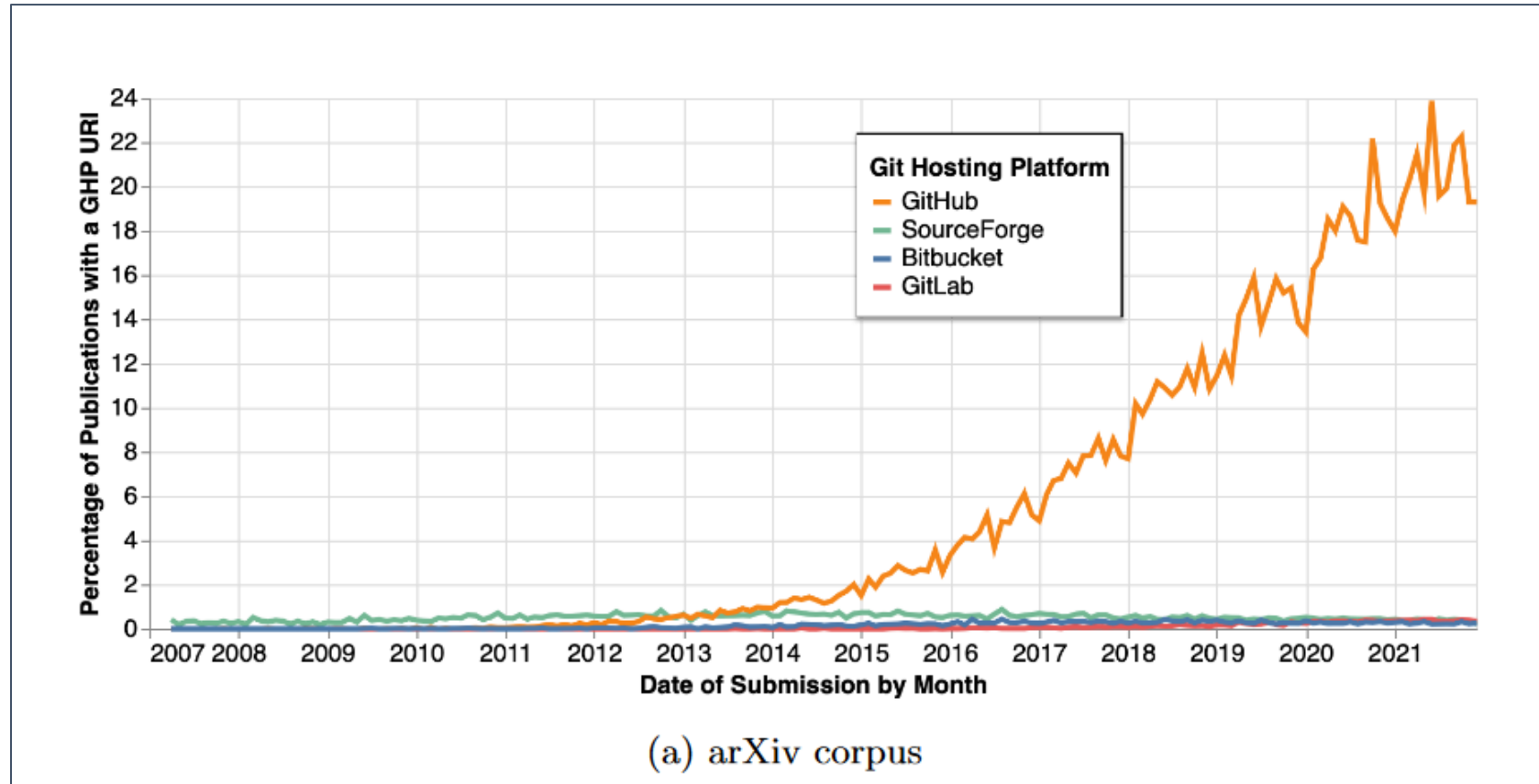
Barba, L. A. (2018). Terminologies for Reproducible Research. *ArXiv:1802.03311 [Cs]*.
<http://arxiv.org/abs/1802.03311>

Quels enjeux pour les logiciels de recherche ?

Photo de Daniel Olah sur Unsplash



Les forges logicielles sont devenues incontournables en recherche ...



🔗 arXiv =
Nearly 2.4 million
scholarly articles in the
fields of physics,
mathematics, computer
science, quantitative
biology, quantitative
finance, statistics,
electrical engineering and
systems science, and
economics

Escamilla, E., Klein, M., Cooper, T., Rampin, V., Weigle, M. C., & Nelson, M. L. (2022). *The Rise of GitHub in Scholarly Publications* (arXiv:2208.04895). arXiv.

<https://doi.org/10.48550/arXiv.2208.04895>

... mais les forges ne sont pas pérennes



Bidding farewell to Google Code

Thursday, March 12, 2015



Bitbucket

Mercurial

gforge.inria.fr was shut down #19757

Open mseri opened this issue on Oct 13, 2021 · 13 comments

mseri commented on Oct 13, 2021 · edited

Many packages some also used by opam are now unavailable. For example mcxs is uninstalleable because cudf is no longer downloadable.

I think they may have moved to gitlab.inria.fr but I didn't check. We should probably prioritise this issue

mseri added bug help wanted labels on Oct 13, 2021

This was referenced on Oct 14, 2021

- Update package overview of camlp5 to move from GForge to GitHub #19776 Merged
- Update package overview of menhir to move from GForge to GitLab #19777 Merged
- Update package overview of mlcuddidl to move from GForge to GitLab #19778 Merged
- Update package overview of coq to move from GForge to GitHub #19779 Merged

2021: Inria's old gforge is unplugged and breaks the Opam build chain for Ocaml

Même les forges institutionnelles ne sont pas des archives : pour des besoins techniques, elles peuvent être amenées à être fermées.

Préserver les logiciels s'apparente à un décathlon

 Complexité d'architecture des logiciels

 Pluralité des sources

“[...] research software is often a thin layer on top of a large number of software dependencies that are developed and maintained outside of academia, so the usual approach based on institutional archives is not sufficient to cover all the software that is relevant for reproducibility of research.”

Di Cosmo, R. (2020). Archiving and Referencing Source Code with Software Heritage. In A. M. Bigatti, J. Carette, J. H. Davenport, M. Joswig, & T. de Wolff (Eds.), *Mathematical Software – ICMS 2020* (Vol. 12097, pp. 362–373). Springer International Publishing. https://doi.org/10.1007/978-3-030-52200-1_36

 Code hosting diaspora

 Enjeux de maintenance

Nowogrodzki, A. (2019). How to support open-source software and stay sane. *Nature*, 571(7763), 133–134. <https://doi.org/10.1038/d41586-019-02046-0>

Ce que Software Heritage (SWH) n'est pas

≠ Un coffre-fort pour les logiciels du passé

≠ Une forge logicielle

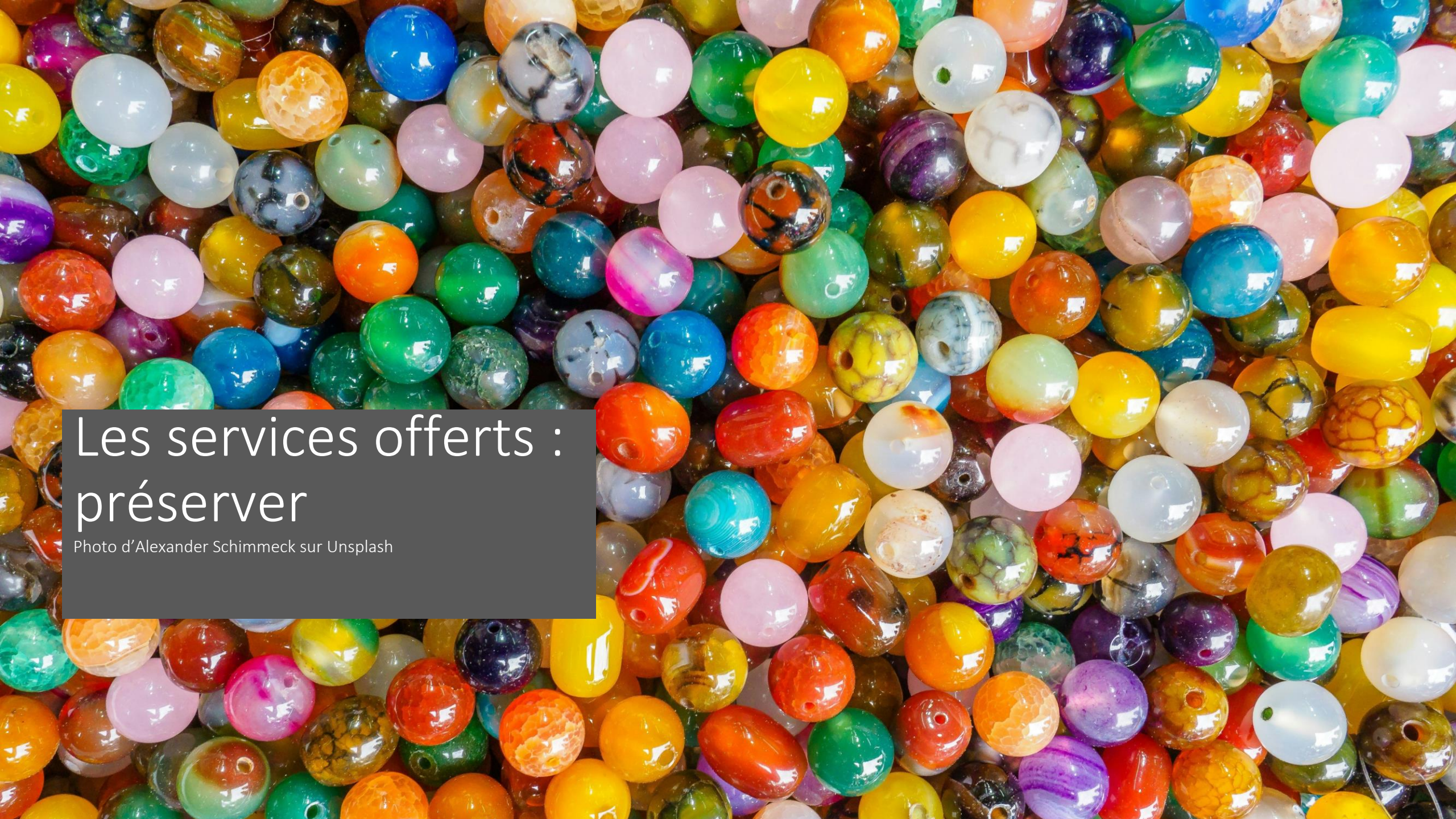
≠ Un doublon des plateformes de dépôt existantes
(notamment, les plateformes d'archivage de données)

≠ Une infrastructure isolée #Partenariats

≠ Un cadre technique isolant #Interopérabilité #Standards

cf. <https://github.com/swhid/specification>

☹ Ce dont on ne
parlera pas
maintenant



Les services offerts : préserver

Photo d'Alexander Schimmeck sur Unsplash

Ce que vous pouvez faire avec Software Heritage

Pas besoin de créer un compte pour utiliser les services:

<https://archive.softwareheritage.org/>



ARCHIVER

 SWH
> 17 milliards de fichiers uniques
> 275 millions de projets

Quels types de logiciels ?

ACCEDER AUX
CONTENUS SUR
LE LONG TERME
Pour les autres
Pour vous-même

RENDRE
TRAÇABLE
PID

CONSULTER VIA
1 POINT D'ACCÈS
CENTRALISÉ

Edit 2024-07-04: Identifier les forges déjà moissonnées, proposer une forge

Bitbucket 2,578,139 origins <	56,983 origins <	git 31,512 origins <																																								
27,377 origins <	debian 138,321 origins <	76,827 origins <																																								
GitHub 210,557,940 origins <	gitiles 10,268 origins <	GitLab 5,606,840 origins ▾																																								
		<table><tbody><tr><td>git.exeter.ac.uk</td><td>git</td><td>75</td><td>🔍</td></tr><tr><td>git.fdn.fr</td><td>git</td><td>8</td><td>🔍</td></tr><tr><td>git.flow3r.garden</td><td>git</td><td>537</td><td>🔍</td></tr><tr><td>git.frama-c.com</td><td>git</td><td>37</td><td>🔍</td></tr><tr><td>git.fs.lmu.de</td><td>git</td><td>45</td><td>🔍</td></tr><tr><td>git.glasklar.is</td><td>git</td><td>101</td><td>🔍</td></tr><tr><td>git.hardenedsbsd.org</td><td>git</td><td>133</td><td>🔍</td></tr><tr><td>git.harrault.fr</td><td>git</td><td>33</td><td>🔍</td></tr><tr><td>git.icinga.com</td><td>git</td><td>102</td><td>🔍</td></tr><tr><td>git.iem.at</td><td>git</td><td>395</td><td>🔍</td></tr></tbody></table>	git.exeter.ac.uk	git	75	🔍	git.fdn.fr	git	8	🔍	git.flow3r.garden	git	537	🔍	git.frama-c.com	git	37	🔍	git.fs.lmu.de	git	45	🔍	git.glasklar.is	git	101	🔍	git.hardenedsbsd.org	git	133	🔍	git.harrault.fr	git	33	🔍	git.icinga.com	git	102	🔍	git.iem.at	git	395	🔍
git.exeter.ac.uk	git	75	🔍																																							
git.fdn.fr	git	8	🔍																																							
git.flow3r.garden	git	537	🔍																																							
git.frama-c.com	git	37	🔍																																							
git.fs.lmu.de	git	45	🔍																																							
git.glasklar.is	git	101	🔍																																							
git.hardenedsbsd.org	git	133	🔍																																							
git.harrault.fr	git	33	🔍																																							
git.icinga.com	git	102	🔍																																							
git.iem.at	git	395	🔍																																							

- <https://archive.softwareheritage.org/>
- Demander l'archivage d'une forge :
<https://archive.softwareheritage.org/add-forge/request/create/>

Edit 2024-07-04: zoom sur l'instance Gitlab Humanum

Publication des dépôts

Les dépôts publics créés dans l'instance GitLab d'Huma-Num sont diffusés sur la plateforme “[Codes sources du secteur public](#)” [Code.gouv.fr](#) développée par [Etalab](#) offrant une visibilité sur les codes sources des organisme publics français. Ils sont aussi archivés dans le cadre de [Software Heritage](#). La publication des dépôts sur [code.gouv.fr](#) implique leur mise sous [licence Ouverte 2.0](#).

Préserver les logiciels : services d'archivage



🤔 Quid d'un logiciel non maintenu?

cf. « Je code : les bonnes pratiques en éco-conception de service numérique à destination des développeurs de logiciels »

Quoi ?

* Contenu

🤔 Quid de l'archivage des **données** dans SWH ? ✓ et ✗

* Niveau de granularité

Comment ? 🛠 et aussi ⚙ (BitBucket, GitHub, Gitlab)

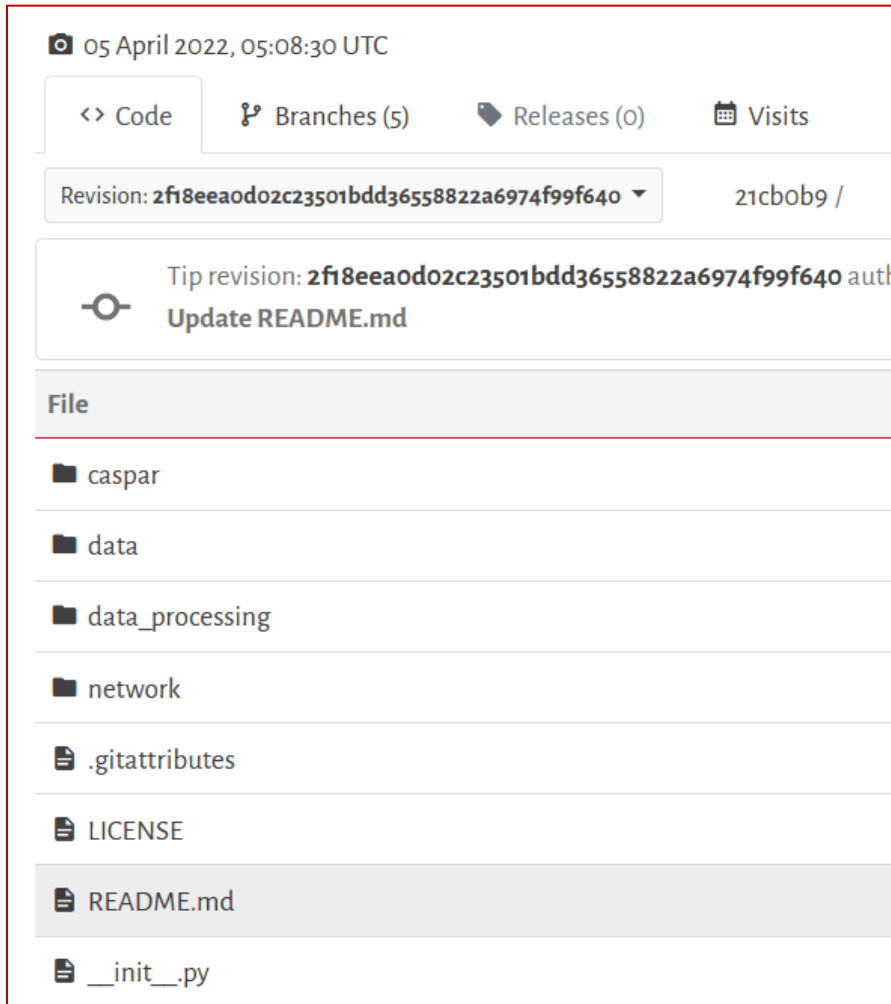
🤔 Quid du **moissonnage** ?

Faire le point :

<https://docs.softwareheritage.org/devel/swh-scanner/index.html>

“Source code scanner to analyze code bases and compare them with source code artifacts archived by Software Heritage.”

Un exemple de code repository



05 April 2022, 05:08:30 UTC

<> Code Branches (5) Releases (0) Visits

Revision: `2f18eea0d02c23501bdd36558822a6974f99f640` 21cbob9 /

Tip revision: `2f18eea0d02c23501bdd36558822a6974f99f640` auth
Update README.md

File

- caspar
- data
- data_processing
- network
- .gitattributes
- LICENSE
- README.md
- __init__.py

<https://archive.softwareheritage.org/swh:1:dir:f4b8e8022b14f7c43feac50ac60b2df0e93091ff;origin=https://github.com/calico/catnap;visit=swh:1:snp:f5146bd01200f5ba3180b0bd03cc6582217e29fb;anchor=swh:1:rev:63e73545f06d3b12142218f34cbf2026ca238a31>

- “This repository contains code for reproducing the results of our eLife paper "Automated, high-dimensional evaluation of physiological aging and resilience in outbred mice." (DOI: <https://doi.org/10.7554/eLife.72664110e72664>)”
- “[data_processing](#) contains the paper-specific code for pre-processing the data from its raw form off the metabolic cage into a per-run / per-trace feature matrix which we use for most of the results in the paper.”

Edit 2024-07-04: Copies et miroirs

- Software Heritage s'appuie sur un réseau de miroirs ainsi que sur des copies pour garantir une pérennité de ses services
- Dans les 2 cas, la stratégie est la même : s'appuyer sur une diversité d'acteurs pour limiter les vulnérabilités techniques
- Microsoft fait partie des sponsors de Software Heritage : une copie est hébergée sur Azure. (Le CINES est également mobilisé) Il s'agit d'une copie de l'archive elle-même. Mais tout ce qui a trait à l'infrastructure elle-même est sur un software stack Software Heritage.
- Les clouds sont utilisés à des fins de rétention de données et d'accessibilité.
- Les sponsors sont soumis au respect de la [charte éthique](#) Software Heritage, au même titre que d'autres utilisateurs.
- Dans le cas des dépôts logiciels dans HAL, l'export vers Software Heritage est optionnel. Chaque déposant peut choisir de ne pas exporter ni les métadonnées, ni le code source.



Les services offerts : rendre identifiable

Photo de Shraga Kopstein sur Unsplash

Différents besoins d'identification appellent différents identifiants



=

Biographie	
Naissance	29 mai 1830 / Vroncourt-la-Côte
Décès	9 janvier 1905 (à Marseille)
Sépulture	Cimetière de Levallois (le 22 janvier 1905)
Nom de naissance	Louise Michel
Pseudonymes	Enjolras, La Vierge Rouge
Nationalité	française
Activités	Enseignante, communarde,

DOI

Enjeux :

- Citation
- « Crédit »

DOI

Enjeu :

- Reproductibilité

Référencer des artefacts logiciels

SWHID

[swh:1:cnt:8819df653b79a79ead8b0dd291e78c8547708e5f;origin=https://github.com/davidB/amethyst-starter-2d;visit=swh:1:snp:a137d958a3259c32e547d7d810feaabee1f3cfd1;anchor=swh:1:rev:a85801125959375c49738deb182d76999715d64e;path=/src/state.rs;lines=34-43](https://github.com/davidB/amethyst-starter-2d?visit=swh:1:snp:a137d958a3259c32e547d7d810feaabee1f3cfd1;anchor=swh:1:rev:a85801125959375c49738deb182d76999715d64e;path=/src/state.rs;lines=34-43)

```
fn handle_event(
    &mut self,
    mut _data: StateData<'_, GameData<'_, '_>>,
    event: StateEvent,
) -> SimpleTrans {
    if let StateEvent::Window(event) = &event {
        // Check if the window should be closed
        if is_close_requested(&event) || is_key_down(&event, VirtualKeyCode::Escape) {
            return Trans::Quit;
        }
    }
}
```

Source code excerpt



Further info in
Resources

```
10 use log::info;
11
12 pub struct MyState;
13
14 impl SimpleState for MyState {
15     // On start will run when this state is initialized. For more
16     // state lifecycle hooks, see:
17     // https://book.amethyst.rs/stable/concepts/state.html#life-cycle
18     fn on_start(&mut self, data: StateData<'_, GameData<'_, '_>>) {
19         let world = data.world;
20
21         // Get the screen dimensions so we can initialize the camera and
22         // place our sprites correctly later. We'll clone this since we'll
23         // pass the world mutably to the following functions.
24         let dimensions = world.read_resource::<ScreenDimensions>().clone();
25
26         // Place the camera
27         init_camera(world, &dimensions);
28
29         // Load our sprites and display them
30         let sprites = load_sprites(world);
31         init_sprites(world, &sprites, &dimensions);
32     }
33
34     fn handle_event(
35         &mut self,
36         mut _data: StateData<'_, GameData<'_, '_>>,
37         event: StateEvent,
38     ) -> SimpleTrans {
39         if let StateEvent::Window(event) = &event {
40             // Check if the window should be closed
41             if is_close_requested(&event) || is_key_down(&event, VirtualKeyCode::Escape) {
42                 return Trans::Quit;
43             }
44
45             // Listen to any key events
46             if let Some(event) = get_key(&event) {
47                 info!("handling key event: {:?}", event);
48             }
49         }
50     }
51 }
```


Obtenir un SoftWare Hash Identifier (SWHID)


The screenshot shows the 'Browse the archive' interface. At the top, there is a search bar with the text 'Enter a SWHID to resolve or keyword(s) to search for in origin URLs'. Below this, the interface displays the details for a specific revision in a GitHub repository: 'https://github.com/baigouy/ScientiFig' with a timestamp of '18 January 2021, 22:58:15 UTC'. The current revision is 'HEAD' with a tip revision of 'b64df8b'. A red vertical label 'Permalinks' is overlaid on the interface. Below the repository information, there is a section titled 'To reference or cite the objects present in the Software Heritage archive, permalinks based on Software Hash Identifiers (SWHIDs) must be used.' This section includes a dropdown menu to select the type of object (directory, revision, snapshot) and a text area containing the generated SWHID: 'swh:1:dir:2f8e74b4142ac1118201435144785e5136b7557b'. Below the text area, there are buttons for 'Copy identifier' and 'Copy permalink', and a checkbox for 'Add contextual information'.

- Quels sont les prérequis ?
- Comment les obtenir ?
 - Via l'interface : <https://archive.softwareheritage.org/>
 - Quels types d'objets ?
- Coût?
- Intervention d'un registre tiers?

Exemples :

Pointe vers une **révision** dans l'historique de développement de Darktable, daté du 16 janvier 2017 : [swh:1:rev:309cf2674ee7a0749978cf8265ab91a60aea0f7d](https://archive.softwareheritage.org/swh:1:rev:309cf2674ee7a0749978cf8265ab91a60aea0f7d)

Pointe vers le **contenu d'un fichier** contenant le texte intégral de la licence GPL3 : [swh:1:cnt:94a9ed024d3859793618152ea559a168bbcbb5e2](https://archive.softwareheritage.org/swh:1:cnt:94a9ed024d3859793618152ea559a168bbcbb5e2)

 Infos détaillées sur les types de SWHID dans « Ressources »

Une réponse aux éditeurs qui veulent des liens

www.nature.com/scientificdata/

Use Case Name, Description	Desired Outcome for JATS XML version 1.3	Desired Outcome for Crossref Metadata depository schema 5.3.1	Notes
<p>Use Case C Software Citation with Software Heritage Persistent Identifier (PID) A registered PID has the desired metadata to support this citation and allows for content negotiation.</p>	<pre><ref id="bib2"> <element-citation publication-type="software"> <person-group person-group-type="author"> <name> <surname> Zhang </surname> <given-names> Martin </given-names> </name> </person-group> <year iso-8601-date="2021">2021</year> <part-title> tabula-muris-senis </part-title> <source> Software Heritage </source> <pub-id pub-id-type="swhid"> swh:1:dir:7dc782970300a97e9bca9038ba34728c857a0638 </ pub-id> </element-citation> </ref></pre>	<p>Tag: //citation/unstructured_citation e.g. <citation key="XXX"> <unstructured_citation> Zhang Martin 2021, tabula-muris-senis, Software Heritage,swh:1:dir:7dc782970300a97e9bca9038ba34728c857a0638 </unstructured_citation> </citation></p>	<p>Con: Non-DOI PIDs are typically not converted to clickable links on journal websites and some journals encourage use of a URL as a preference.</p>

Stall, S., Bilder, G., Cannon, M., Chue Hong, N., Edmunds, S., Erdmann, C. C., Evans, M., Farmer, R., Feeney, P., Friedman, M., Giampoala, M., Hanson, R. B., Harrison, M., Karaiskos, D., Katz, D. S., Letizia, V., Lizzi, V., MacCallum, C., Muench, A., ... Clark, T. (2023). Journal Production Guidance for Software and Data Citations. *Scientific Data*, 10(1), Article 1. <https://doi.org/10.1038/s41597-023-02491-7>


Permalinks

Copy identifier Copy permalink


<https://archive.softwareheritage.org/swh:1:dir:7dc782970300a97e9bca9038ba34728c857a0638>

Un exemple parmi d'autres d'une vie au-delà des DOI : un article qui contient une version permalink du SWHID

article is itself a Maneage'd project (project commit 313db0b). Appendices—Two comprehensive appendices that review the longevity of existing solutions are available as supplementary “Web extras,” which are available in the IEEE Computer Society Digital Library at <http://doi.ieeecomputersociety.org/10.1109/MCSE.2021.3072860>. Reproducibility—All products available in zenodo.4913277, the Git history of this paper's source is at git.maneage.org/paper-concept.git, which is also archived in Software Heritage:
swh:1:dir:33fea87068c1612daf011f161b97787b9a0df39fk. Clicking on the SWHIDs in the digital format will provide more “context” for same content.



Akhlaghi, M., Infante-Sainz, R., Roukema, B. F., Khellat, M., Valls-Gabaud, D., & Baena-Gallé, R. (2021). Toward Long-Term and Archivable Reproducibility. *Computing in Science & Engineering*, 23(3), 82–91.
<https://doi.org/10.1109/MCSE.2021.3072860>



SWH contribue à enrichir
votre workflow de recherche
HAL | Graphics Replicability Stamp Initiative | Guix

Photo d'Alina Grubnyak sur Unsplash



Quoi ? « L'archive ouverte multidisciplinaire choisie par l'ensemble de la communauté scientifique et universitaire française pour la diffusion des savoirs.

HAL est à la fois une application, un entrepôt unique, et une plateforme mutualisée pour les archives institutionnelles, les archives ouvertes thématiques et les thèses en accès ouvert. »

<https://www.ccsd.cnrs.fr/hal/>

Pourquoi ? « La plateforme garantit l'accès et la préservation à long terme de ces documents et propose un ensemble de services qui contribuent à leur valorisation. »

Déposer un logiciel dans HAL pour :
compléter CV et rapports, être cité.e en
tant qu'auteur de logiciel
(Et aussi, citer les logiciels des autres)



=

Biographie	
Naissance	29 mai 1830 ✎ Vroncourt-la-Côte ✎
Décès	9 janvier 1905 ✎ (à 74 ans) Marseille ✎
Sépulture	Cimetière de Levallois-Perret (depuis le 22 janvier 1905) ✎
Nom de naissance	Louise Michel ✎
Pseudonymes	Enjolras, La Vierge Rouge ✎
Nationalité	française ✎
Activités	Enseignante, communarde, ✎

HAL-ID



SWHID

Photo de [Immo Wegmann](#) sur [Unsplash](#)

Éléments de contexte : des pratiques actuelles de citation très hétérogènes

- Le logiciel n'est pas encore considéré comme une source qui doit être citée, au même titre qu'un article ou qu'un ouvrage
- Les pratiques sont peu standardisées actuellement : simple mention du nom sans référence complémentaire, ajout d'un lien vers un site ou une forge, citation de l'article qui décrit le logiciel.
- Avantages de citer l'article qui décrit le logiciel: facilite le décompte des citations et permet d'avoir une meilleure idée de l'impact dans le monde académique
- Mais il n'y a pas toujours d'article décrivant le logiciel, notamment quand celui-ci est porté par une communauté hors du secteur de la recherche et cette solution a ses limites

L'ascenseur émotionnel de la citation de logiciel : les limites de la citation de l'article sur le logiciel

“Yet, using citations to articles can run counter to the need to identify and find the software itself, particularly because **the publication citations remain static while software changes, including changing name.**” 🤔

Howison, J., & Bullard, J. (2016). Software in the scientific literature: Problems with seeing, finding, and using software mentioned in the biology literature. *Journal of the Association for Information Science and Technology*, 67(9), 2137–2155.
<https://doi.org/10.1002/asi.23538>

😱 The authors had cited an OGSA-DAI paper that should have meant they were using a version of the software between OGSA-DAI 1 and 6.
😞 Later in their paper, the authors mentioned a component that was specific to OGSA-DAI versions 2.5 to 6.
😞 However, the authors then talked of another component and a toolkit, which was only available with a completely different version of the software.

Without my highly detailed knowledge of the OGSA-DAI project, it would have been impossible to determine what software was used.”

Jackson, M. (s.d.). *How to cite and describe software*. Software Sustainability Institute. <https://www.software.ac.uk/how-cite-software>

Il y a des domaines de recherche où citer directement le logiciel fait la différence : permettre une identification précise

“Examples of areas where the software used directly impacts upon the results, and so needs to be described, includes

numerical modelling or simulations, usability evaluations, performance evaluations of algorithms (where the evaluations are done using implementations of the algorithms), or

research using software that does some form of automated analysis (e.g. image analysis or optical character recognition). ”


Jackson, M. (s.d.). *How to cite and describe software*. Software Sustainability Institute. <https://www.software.ac.uk/how-to-cite-software>

Un impact du logiciel difficile à mesurer, en raison de la faible standardisation des pratiques de citation et de la nature évolutive du logiciel

“Due to the diverse forms that a mention of software can take in a scholarly work, fully automated extraction of references is far from being sound and complete (Howison & Bullard, 2015), and there are different experiments ongoing.”

EOSC Executive Board & EOSC Secretariat. (2020). *Scholarly infrastructures for research software. Report from the EOSC Executive Board Working Group (WG) Architecture Task Force (TF) SIRS*. European Commission. Directorate General for Research and Innovation. <https://data.europa.eu/doi/10.2777/28598>

Des fonctionnalités d'export qui facilitent la citation


Consulter sur  Software Heritage

Logiciel Année : 2019

A sound and complete axiomatization of the equational theory of Mealy machines.

Cécilia Pradic (1, 2)

[Afficher plus de détails](#)

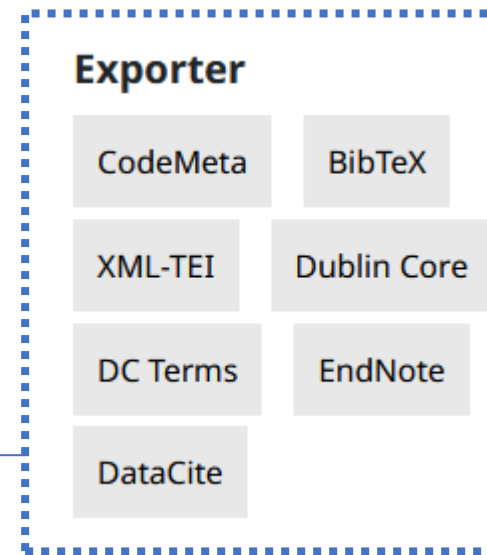
 1 ENS de Lyon - École normale supérieure de Lyon
2 MIMUW - Faculty of Mathematics, Informatics, and Mechanics [Warsaw]

Résumé **en**

Mots clés **en**

Coq transducer interactive theorem proving

This small development in the Coq proof assistant introduces a term language and a basic equational theory for Mealy machines, i.e. finite-state letter-to-letter transducer. Soundness and completeness are proved for the equational theory



<https://hal.science/hal-02155786>

```
@softwareversion{pradic:hal-02155786v1,
  TITLE = {{A sound and complete axiomatization of the equational theory of Mealy machines.}},
  AUTHOR = {Pradic, C{\e}cilia},
  URL = {https://inria.hal.science/hal-02155786},
  NOTE = {},
  YEAR = {2019},
  MONTH = Jun,
  SWHID = {swh:1:dir:d26d7ad1514340934a5b546032e21ad2fcc87441;origin=https://hal.archives-ouvertes.fr/hal-02155786;visit=swh:1:snp:88a3a6cafd6d6d842ba0c086187015a1605f0cf0;anchor=swh:1:rev:dacff8b9c3574cd3fcb023c485209c72abd8efef;path=/},
  VERSION = {0.1},
  LICENSE = {GNU Lesser General Public License v3.0 only},
  KEYWORDS = {Coq ; transducer ; interactive theorem proving},
  FILE = {https://inria.hal.science/hal-02155786/file/mealy-0.1.tar.gz},
  HAL_ID = {hal-02155786},
  HAL_VERSION = {v1},
}
```

2 modalités de dépôts dans HAL selon les modalités de développement

Qui peut faire le dépôt ?



Tutoriels et documentation dans « Ressources »

Le logiciel est développé hors forge

- Préparer des fichiers pour le dépôt
- Compléter les métadonnées du formulaire HAL
 - Export vers SWH possible

Avantage majeur ?

Le logiciel est développé dans une forge

- Cas 1 : Le logiciel est archivé dans SWH : copier-coller le SWHID pour alimenter les données HAL, vérifier les métadonnées
- Cas 2 : Le logiciel n'est pas archivé dans SWH : effectuer l'archivage (cf. diapos et « Ressources »), et reprendre les étapes du cas 1

Avantage majeur ?

😬 Il existe plusieurs niveaux de description : métadonnées intrinsèques et extrinsèques ...

README.md

Parmap in a nutshell

Parmap is a minimalistic library allowing to exploit multicore architecture for OCaml programs with minimal modifications: if you want to use your many cores to accelerate an operation which happens to be a map, fold or map/fold (map-reduce), just use Parmap's `parmap`, `parfold` and `parmapfold` primitives in place of the standard `List.map` and friends, and specify the number of subprocesses to use by the optional parameter `~ncores`.

See the `example` directory for a couple of running programs.

DO'S and DONT'S


Parmap is *not* meant to be a replacement for a full fledged implementation of

parallelism skeletons (map, reduce, pipe, ...), as described in scientific literature since the end of the 1990s. It is not a specific implementation by Google engineers, but rather a simple meant, instead, to allow you to quickly leverage the power of your extra cores, when handling some heavy computations.

The principle of parmap is very simple: write your computation primitives, `map`, `fold`, and `mapfold`, your OCaml code, and use subprocesses (you choose the `n`), and each child processes the `1/n` of the data, in chunks of a size you choose, through a shared memory area to the parent process. When all the children have terminated, and the data is ready to be processed, the parent process will return the result.

AUTHORS

```
1 This code started as a two day exercise done at Irill in July 2011, by
2 Marco Danelutto <marcod@di.unipi.it> and Roberto Di Cosmo <roberto@dicosmo.org>,
3 that produced the architecture now contained in simplemapper.
4
5 It has evolved significantly after, and it now contains a particularly efficient
6 version of the map code on float arrays, incorporating ideas from Paul Vernaza
7 <vernaza@andrew.cmu.edu>, and that works thanks to smart code and significant
8 insight from Jerome Vouillon <Jerome.Vouillon@pps.jussieu.fr>.
9
10 Pietro Abate <Pietro.Abate@pps.jussieu.fr> provided autoconf and ocamlbuild support.
11
12 Detailed contributor history is contained in the git repository available at
13
14 https://github.com/rdicosmo/parmap
15
```

Format du dépôt	Fichier
Type de document	Logiciel
Titre	en Parmap
Résumé	en Parmap is a minimalistic library allowing to exploit multicore architecture for OCaml programs with minimal modifications: if you want to use your many cores to accelerate an operation which happens to be a map, fold or map/fold (map-reduce), just use Parmap's <code>parmap</code> , <code>parfold</code> and <code>parmapfold</code> primitives in place of the standard <code>List.map</code> and friends, and specify the number of subprocesses to use by the optional parameter <code>~ncores</code> .
Auteur(s)	Roberto Di Cosmo ^{1,2} , Marco Danelutto ³ 1 DGD-I - Direction générale déléguée à l'innovation (50190) - Institut National de Recherche en Informatique et en Automatique 2 - Software Heritage (498515) - France 3 - University of Pisa - Università di Pisa (366408) - Lur
Code Repository	https://github.com/rdicosmo/parmap.git
Date de production/écriture	2011-07-18
Version	1.2.4
Etat du développement	active
Licences	• GNU Library General Public License v2 only
Platform/OS	• Linux • MacOS
Domaine	• Informatique (cs)/Langage de programmation (cs.PL)
Langage de programmation	• OCaml
Classification ACM 2012	• acm2012_ACM2012/Computing methodologies/Parallel computing
Publications associées	• 10.1016/j.procs.2012.04.202 
Mots-clés	en parallel computing, functional programming, OCaml
SWHID	swh:1:dir:ec88e5b901c034d5a91aa133e824d65cff3788a3;orcid:0000-0001-9088-4000;anchor=swh:1:rev:db44dc9cf7a6af7b56d8ebda8c75be3375d



... Et il existe surtout des outils pour faciliter la production de métadonnées extrinsèques

Pour les R addicts : CodemetaR

- Automatically extract as much metadata as possible using the DESCRIPTION file
- Facilitate the addition of further metadata fields into a codemeta.json file

<https://docs.ropensci.org/codemetar/>

Pour les Pythoniennes et les Pythoniens : CodemetaPy

- Allows chaining to successively update a metadata description based on multiple sources.
- It can also read and manipulate existing codemeta.json files as well as parse simple AUTHORS/CONTRIBUTORS files

<https://github.com/proycon/codemetapy>



Alternative : CodeMetaGenerator

Most fields are optional. Mandatory fields will be highlighted when generating Codemeta.

The software itself Name My Software the software title Description My Software computes ephemerides and orbit propagation. It has been developed from early '80. Creation date YYYY-MM-DD First release date YYYY-MM-DD License(s) from SPDX licence list Current version of the software Version number 1.0.0 Release date YYYY-MM-DD Download URL https://example.org/MySoftware.tar.gz Release notes Change log: this and that; Bugfixes: that and this.	Discoverability and citation Unique identifier 10.151.xxxxx such as ISBNs, GTIN codes, UUIDs etc. http://schema.org/identifier Application category Astronomy Keywords ephemerides, orbit, astronomy Funding PRA_2018_73 grant funding software development Funder Università di Pisa organization funding software development Authors and contributors can be added below	Development community / tools Code repository git+https://github.com/You/RepoName.git Continuous integration https://travis-ci.org/You/RepoName Issue tracker https://github.com/You/RepoName/issues Related links Additional Info Reference Publication https://doi.org/10.1000/xyz123 Development Status see www.repostatus.org for details Is part of http://The.Bigger.Framework.org	Run-time environment Programming Language C#, Java, Python 3 Runtime Platform .NET, JVM Operating System Android 1.6, Linux, Windows, macOS Other software requirements Python 3.4 https://github.com/psf/requests
--	---	--	--

Et les données du README?

CodeMeta Generator
<https://codemeta.github.io/codemeta-generator/>
JSON-LD format

On peut définir l'effort raisonnable en termes de description : que citer

Critère de selection :

“You should cite software [...] that has a significant impact on the research outcome presented in your work, or on the way the research has been conducted. If the research you are presenting is not repeatable without a piece of software, then you should cite the software.”

Chue Hong, N. P., Allen, A., Gonzalez-Beltran, A., de Waard, A., Smith, A. M., Robinson, C., Jones, C., Bouquin, D., Katz, D. S., Kennedy, D., Ryder, G., Hausman, J., Hwang, L., Jones, M. B., Harrison, M., Crosas, M., Wu, M., Löwe, P., Haines, R., ... Pollard, T. (2019). *Software Citation Checklist for Authors (0.9.0)*. FORCE11 Software Citation Implementation Working Group. <https://doi.org/10.5281/zenodo.3479199>

Typologie et identification :

1/ Typologie, outre le logiciel : des scripts ; un framework ; une plateforme ; des packages, des plugins, des modules, des bibliothèques

2/ “[...] the software used might not be a release with a version number but a check-out from a source code repository, in which case, **the version could be described in terms of a combination of a repository URL, check-out date, branch or tag name, or a revision number.**”

Jackson, M. (s.d.). *How to cite and describe software*. Software Sustainability Institute. <https://www.software.ac.uk/how-cite-software>

« Mais j'utilise déjà Zenodo pour décrire mes logiciels » (Zenodo ≠ SWH)


Depuis la notice Zenodo

- Ajouter le SWHID dans la [description](#)


Depuis le code repository, dans le README


- Ajouter le DOI Zenodo du logiciel
- Ajouter éventuellement les DOI vers les jeux de données

🤖 Qu'est-ce que Zenodo? Pourquoi et dans quels cas l'utiliser ? Quels sont ses avantages et ses inconvénients ?
<https://libraries.ou.edu/impact-challenge-chapter/make-your-data-discoverable>
<https://zenodo.org/records/802100>

 Software repository

 Paper review

 Download paper

 Software archive

Le JOSS utilise
Zenodo



Quoi ? Graphics Replicability Stamp Initiative n'est pas une revue mais un service complémentaire pour attester la reproductibilité du code source. L'auteur qui le souhaite envoie sa demande pour un tampon après acceptation de sa publication dans une revue.

<https://www.replicabilitystamp.org/>

Pourquoi ? “[...] to allow scientists and practitioners to immediately benefit from state-of-the-art research results, without spending months re-implementing the proposed algorithms and trying to find the right parameter values.”

Augmenter son taux de citation en soumettant un code source

Revue partenaires de la GRSI

[ACM Transaction on Graphics](#)

[IEEE Transaction on Visualization and Computer Graphics](#)

[Wiley Computer Graphics Forum](#)

[Elsevier Computers & Graphics](#)

[Elsevier Computer Aided Geometric Design](#)

[Elsevier Computer-Aided Design](#)

Tout article accepté en 2017 ou après dans une des revues est éligible

“Papers with the Replicability Stamp are cited **15% more** than average one year after publication, and **33% more** two years after publication. [Calculated based on Clarivate Web Of Science (WoS) citation counts in 2020]”

Une procédure de soumission allégée et un code archivé dans SWH si validé

- general information on the submission
- a representative image
- a link to a public git repository with the source code
- instructions on how to compile and replicate the results

“The code should

compile on a vanilla installation of one of the major operating systems (Linux, MacOSX, or Windows),

have a license that allows non-commercial usage, and only depend on libraries that are free for academic or research purposes.”



<https://www.replicabilitystamp.org/index.html#https-github-com-interactivecomputergraphics-splishsplash-v2>



Quoi ? Un gestionnaire de paquets universel, soit « une collection d’outils logiciels qui automatise le processus d’installation, de mise à jour, de configuration et de suppression des logiciels sur le système d’exploitation. »

<https://gricad-doc.univ-grenoble-alpes.fr/hpc/softenv/guix/#whatisguix>

<https://guix.gnu.org/>

Pourquoi ? Les variations d’environnements peuvent entraver la reproductibilité des travaux. Guix offre un environnement stable.

Courtès, L. (2016). *Gestion de paquets sûre et flexible avec GNU Guix*. <https://people.bordeaux.inria.fr/lcourtes/guix-glmf-201606.pdf>

« Avant Guix, l'installation de logiciels scientifiques était nécessairement ad hoc.

Les équipes [de recherche] compilaient leurs logiciels et les liaient statiquement, en espérant que ça ne change jamais (parce que la gestion des environnements logiciels était pratiquement impossible).

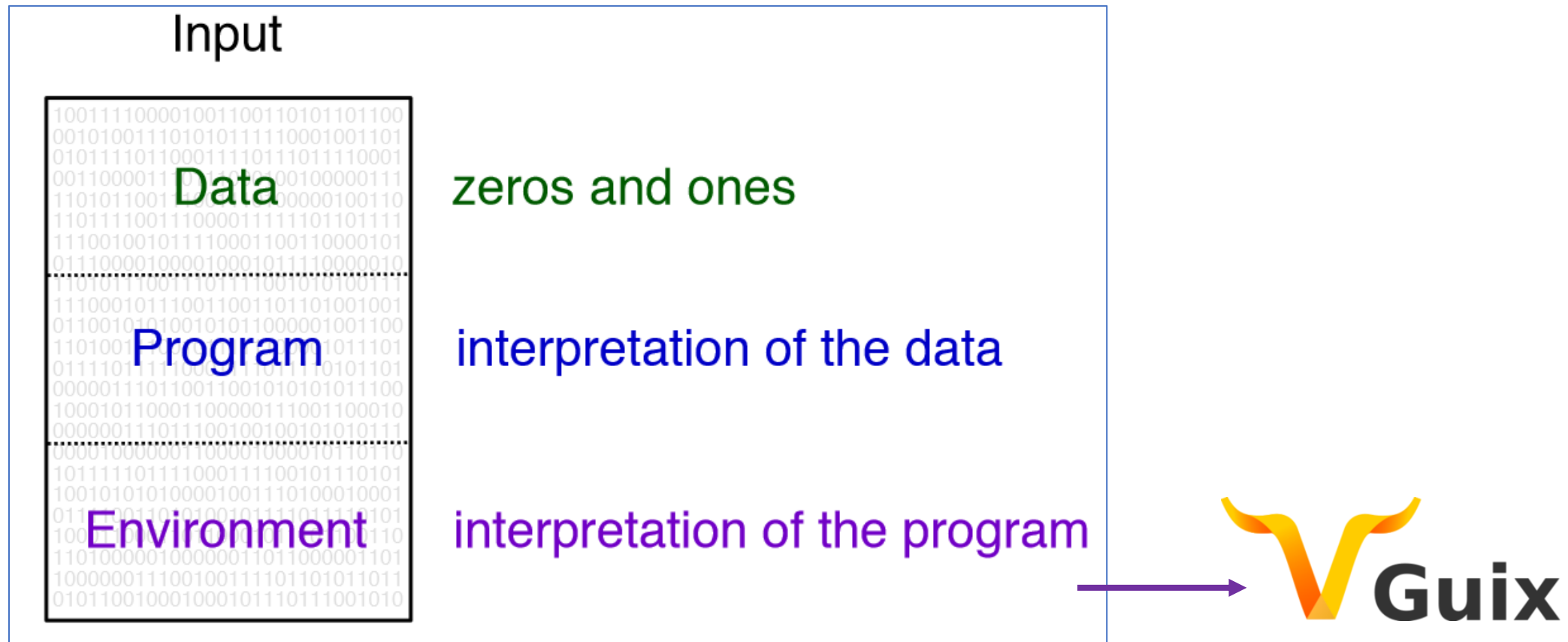
Dorénavant on peut non seulement gérer un environnement par équipe de façon fiable, mais on peut aussi avoir des environnements à tous les niveaux : équipe de recherche, projet, usager, *workflow*, etc. »

Ricardo Wurmus, administrateur système de la *Scientific Bioinformatics Platform* du *Max Delbrück Institute*

En savoir plus, comprendre la plus-value de Guix/conteneurs :

<https://www.inria.fr/fr/guix-un-logiciel-libre-pour-la-reproductibilite-des-sciences-en-hpc>


Le code source constitue une brique majeure dans l'effort de reproductibilité des environnements



Hinsen, K. (2019, May 23). *Enjeux et défis de la recherche reproductible*. Journée Système : La reproductibilité en pratique, méthodes et outils, Villeurbanne.

https://indico.mathrice.fr/event/165/contributions/299/attachments/336/395/Aramis2019_HINSEN.pdf

Quand l'URL que vous avez demandée n'est plus disponible,
SWH prend le relais

 « L'idée est de voir chaque paquet comme une valeur immuable, résultat de l'application d'une fonction de compilation à un ensemble d'arguments » (Courtès, L. 2016)


Guix [package definition](#)

=


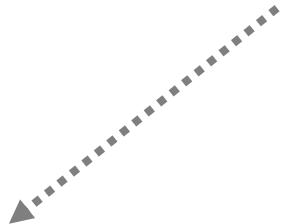
URL du code source du package, hash attendu pour le fichier téléchargé, dépendances du package, procédure de construction du paquet

https://guix.gnu.org/manual/fr/html_node/Definition-des-paquets.html

https://guix.gnu.org/manual/fr/html_node/reference-de-origin.html

 L'URL du code source ne fonctionne plus ?

Disparition, délocalisation

 
https://guix.gnu.org/manual/fr/html_node/Invoquer-guix-lint.html

A top-down view of several black plastic pots containing young green seedlings in dark soil. The pots are arranged in a grid-like pattern, and the seedlings are at various stages of growth. A semi-transparent grey box is overlaid on the right side of the image, containing the title and photo credit.

Rejoindre la communauté de Software Heritage

Photo de Markus Spiske sur Unsplash

S'informer, obtenir de l'aide, faire des propositions

<https://sympa.inria.fr/sympa/info/swh-science>

<https://www.softwareheritage.org/newsletter/>

<https://www.softwareheritage.org/blog>

<https://matrix.to/#/#swh:matrix.org> (#swh-devel for development discussions)

info@softwareheritage.org

Solliciter un.e membre du réseau des ambassadeurs Software Heritage

<https://www.softwareheritage.org/ambassadors/>

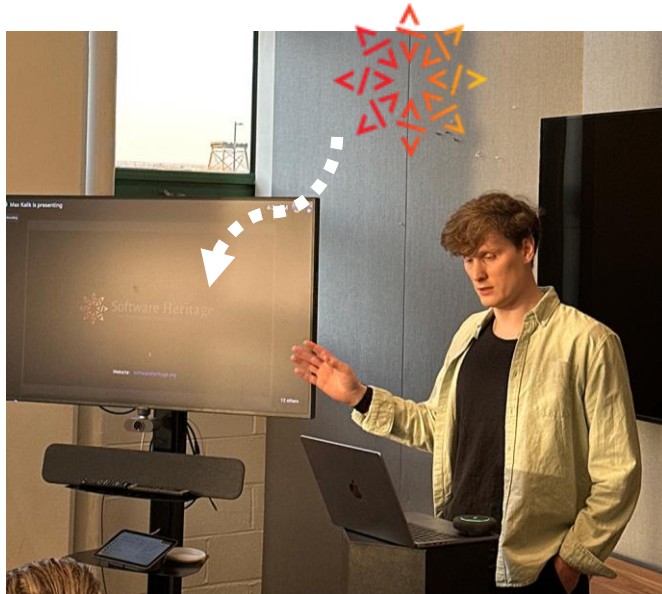
4 domaines :

- Academia
- Cultural Heritage
- Industry
- Public Administration

Types d'interventions

Coût

Devenir un.e membre du réseau des ambassadeurs Software Heritage



Max Kalik au iOSoho meetup (2023)



“Services should assign PIDs to data descriptions, data and complementary materials (e.g., digital object identifiers (DOI)), **software (DOI, Software Heritage ID (SWHID)¹⁴**), authors (open researcher and contributor IDs (ORCID)) and associated research resources (RRIDs¹⁵).”

Sandström, Malin, Abrams, M., Bjaalie, J. G., Hicks, M., Kennedy, D. N., Kumar, A., Poline, J.-B., Roy, P. K., Tiesinga, P., Wachtler, T., & Goscinski, W. J. (2022). **Recommendations for repositories and scientific gateways from a neuroscience perspective**. *Scientific Data*, 9(1), Article 1. <https://doi.org/10.1038/s41597-022-01334-1>



Blog post de Simon Phipps

Merci !



Sélection de ressources

Archiver son code dans Software Heritage



Options manuelles

Save Code Now : <https://archive.softwareheritage.org/save/>

Extensions de navigateurs : <https://www.softwareheritage.org/browser-extensions/>



Options automatisées

GitHub action : <https://github.com/marketplace/actions/save-to-software-heritage>

Webhook (BitBucket, GitHub, Gitlab):

https://docs.softwareheritage.org/user/save_code_now/webhooks/index.html

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



Archiver un ensemble de projets

Add Forge Now: <https://archive.softwareheritage.org/add-forge/request/create/>

Archiver son code dans Software Heritage : rendez les releases !



Je ne retrouve pas toutes les releases dans la version archivée.

Your repository has been saved in full.

What you are witnessing is a terminological difference between what platforms like GitHub calls “releases” (any non annotated git tag) and what we call “releases” (a node in the Merkle tree, which corresponds to a git annotated tag).

This is a common issue, as you can see for example in [this discussion thread](#).

[...]

If you want a release to appear in our web interface you should create your tags using “git tag -a”, instead of simply “git tag”, or create the release directly on the code hosting platform, that uses the proper “git tag -a” behind the scenes, and then archive your repository again.

Ces contenus sont-ils adaptés à l'archivage dans Software Heritage ?

➔ J'utilise Software Heritage

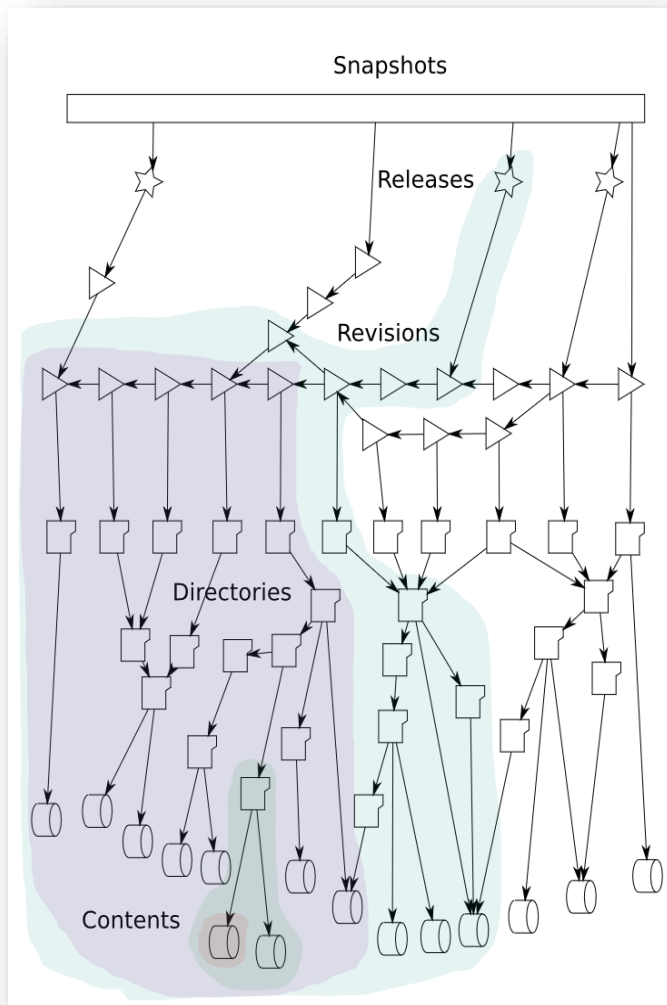
- Il s'agit des fichiers de code source
- Les données sont indissociables du logiciel : elles ont servi à son développement, elles permettent son fonctionnement

↶ J'utilise une autre plateforme

- Il s'agit d'une donnée de sortie ou d'une donnée d'entrée non liée au développement ou au fonctionnement du logiciel
- Ce sont des binaires

<https://www.softwareheritage.org/save-and-reference-research-software/>

Les types d'artefacts identifiables avec un SWHID



<https://docs.softwareheritage.org/devel/swh-model/persistent-identifiers.html#id10>

Snapshot	Branches (ou suites) à un moment donné (Git does not have a corresponding object type) Snapshot- a capture of the complete situation in a software repository, including branches, releases and all the development history.	A snapshot of the entire Darktable Git repository taken on 4 May 2017 from GitHub: swh:1:snp:c7c108084bc0bf3d81436bf980b46e98bd338453
Release	// git release hash a specific version can be shared as a release on a package manager or as a tar file on a website, it can also be a tagged revision in a version control system	Darktable release 2.3.0, dated 24 December 2016 swh:1:rel:22ece559cc7cc2364edc5e5593d63ae8bd229f9f
Revision	// git commit hash a specific point in development history- in a version control system, this is the mechanism that captures the modifications in each iteration during the software development.	A commit in the development history of Darktable, dated 16 January 2017, that added undo/redo supports for masks: swh:1:rev:309cf2674ee7a0749978cf8265ab91a60aea0f7d
Directory	// git tree hash	A directory containing the source code of the Darktable photography application as it was at some point on 4 May 2017: swh:1:dir:d198bc9d7a6bcf6db04f476d29314f157507d505
Content	The SHA1 of a byte sequence obtained by juxtaposing the ASCII string "blob" (without quotes), a space, the length of the content as decimal digits, a NULL byte, and the actual content of the file	swh:1:cnt:94a9ed024d3859793618152ea559a168bbcbb5e2;path=/

Déposer son code dans HAL : documentation

“Create software deposit in HAL: User guide and best practices”:

<https://hal.archives-ouvertes.fr/hal-01872189>

https://scienceouverte.univ-grenoble-alpes.fr/wp-content/uploads/2023/10/webinaire_diffusionCode_octobre2023.pdf

Tutoriels: <https://youtube.com/playlist?list=PLD2VqrZz2-u3bOWtoCoBlh5Flt6iYXsq3>

On peut définir l'effort raisonnable en termes de description : noyau minimal d'informations

- le nom du projet de la manière la plus précise possible (nom du package, de la bibliothèque)
- la date de mise à disposition de la version ou à défaut, la date d'utilisation du logiciel par l'utilisateur
- l'auteur
- la localisation (//origin)
- l'identifiant : SWHID
- des informations contextuelles, en cas d'homonymie : « Additional Context/Description beyond acronym or code name (e.g., Longhorn pipeline scripts for reducing data vs Longhorn) » Stall, S., Bilder, G., Cannon, M., Chue Hong, N., Edmunds, S., Erdmann, C. C., Evans, M., Farmer, R., Feeney, P., Friedman, M., Giampoala, M., Hanson, R. B., Harrison, M., Karaiskos, D., Katz, D. S., Letizia, V., Lizzi, V., MacCallum, C., Muench, A., ... Clark, T. (2023). Journal Production Guidance for Software and Data Citations. *Scientific Data*, 10(1), Article 1. <https://doi.org/10.1038/s41597-023-02491-7>

Chue Hong, N. P., Allen, A., Gonzalez-Beltran, A., de Waard, A., Smith, A. M., Robinson, C., Jones, C., Bouquin, D., Katz, D. S., Kennedy, D., Ryder, G., Hausman, J., Hwang, L., Jones, M. B., Harrison, M., Crosas, M., Wu, M., Löwe, P., Haines, R., ... Pollard, T. (2019). *Software Citation Checklist for Authors (0.9.0)*. FORCE11 Software Citation Implementation Working Group. <https://doi.org/10.5281/zenodo.3479199>

Sur les pratiques de citation de logiciels

Katz, D. S., Chue Hong, N. P., Clark, T., Muench, A., Stall, S., Bouquin, D., Cannon, M., Edmunds, S., Faez, T., Feeney, P., Fenner, M., Friedman, M., Grenier, G., Harrison, M., Heber, J., Leary, A., MacCallum, C., Murray, H., Pastrana, E., ... Yeston, J. (2021). Recognizing the value of software: A software citation guide. *F1000Research*, 9, 1257. <https://doi.org/10.12688/f1000research.26932.2>

Chue Hong, N. P., Allen, A., Gonzalez-Beltran, A., de Waard, A., Smith, A. M., Robinson, C., Jones, C., Bouquin, D., Katz, D. S., Kennedy, D., Ryder, G., Hausman, J., Hwang, L., Jones, M. B., Harrison, M., Crosas, M., Wu, M., Löwe, P., Haines, R., ... Pollard, T. (2019). *Software Citation Checklist for Authors (0.9.0)*. FORCE11 Software Citation Implementation Working Group. <https://doi.org/10.5281/zenodo.3479199>

Smith, A. M., Katz, D. S., & Niemeyer, K. E. (2016). Software citation principles. *PeerJ Computer Science*, 2, e86. <https://doi.org/10.7717/peerj-cs.86>

Le biblatex-software package

<https://ctan.org/pkg/biblatex-software>

- ❑ supports SWHID natively
- ❑ integrated in CTAN and TeXLive
- ❑ works out of the box in Overleaf
- ❑ integrated in [the ACM article style](#)

[Documentation: http://mirrors.ctan.org/macros/latex/contrib/biblatex-contrib/biblatex-software/software-biblatex.pdf](http://mirrors.ctan.org/macros/latex/contrib/biblatex-contrib/biblatex-software/software-biblatex.pdf)

Tutoriel : <https://youtu.be/UhQCeAj9yKM>

Comprendre le contexte national | PNSO

2021-2024 : Deuxième plan national pour la science ouverte (PNSO) ; Axe 3 « Ouvrir et promouvoir les codes sources produits par la recherche »

<https://www.ouvrirelascience.fr/deuxieme-plan-national-pour-la-science-ouverte-pnso/>

3 objectifs :

- Valoriser et soutenir la diffusion sous licence libre des codes sources issus de recherches financées sur fonds publics.
- Mettre en valeur la production des codes sources de l'enseignement supérieur, de la recherche et de l'innovation.
- Définir et promouvoir une politique en matière de logiciels libres.

SWH est inscrit
dans le PNSO 2

Comprendre le contexte national | ANR

Recommandations 2023 de l'ANR (p.17) :

https://anr.fr/fileadmin/aap/2023/aapg-2023-V2.0_MAJ20220921.pdf

Enfin, conformément au 2^{ème} Plan national pour la science ouverte, L'ANR recommande que les logiciels développés durant le projet soient mis à disposition sous une licence libre³⁰ et que les codes sources soient stockés dans l'archive Software Heritage³¹ en indiquant la référence au financement ANR.

Comprendre le contexte international

- 2019 : Appel de Paris, Unesco <https://en.unesco.org/news/experts-call-greater-recognition-software-source-code-heritage-sustainable-development>
- 2018-2021 : recommandations de l'Unesco pour la science ouverte <https://unesdoc.unesco.org/ark:/48223/pf0000378381.locale=en>
- 2020 : rapport SIRS, European Open Science Cloud <https://data.europa.eu/doi/10.2777/28598>

Les partenariats avec des éditeurs

eLife

Ipol

The source code of LabNet is available over the [GitHub repository](#) under a GPL-3.0 license.

Data of the performance measurements, the source code for Autopilot, Bpod and pyControl tasks, and the source code for the graphs are also accessible via the [GitHub repository](#), (copy archived at swh:1:rev:d52e52c51e3f7c5b0e12f95829b8cf4886bb3379; [Schatz and Winter, 2022](#)).

Biochemistry and Chemical
Biology
Cancer Biology
Cell Biology
Chromosomes and Gene
Expression
Computational and Systems
Biology
Developmental Biology
Ecology
Epidemiology and Global
Health
Evolutionary Biology
Genetics and Genomics

Immunology and
Inflammation
Medicine
Microbiology and Infectious
Disease
Neuroscience
Physics of Living Systems
Plant Biology
Stem Cells and Regenerative
Medicine
Structural Biology and
Molecular Biophysics
<https://elifesciences.org/>



An Analysis and Implementation of the Shape Preserving Local Histogram Modification Algorithm

Jose-Luis Lisani

[article](#) [demo](#) [archive](#)

published • 2018-12-07

reference • JOSE-LUIS LISANI, *An Analysis and Implementation of the Shape Preserving Local Histogram Modification Algorithm*, Image Processing On Line [CR](#), 8 [CR](#) (2018), pp. 408–434. <https://doi.org/10.5201/ipol.2018.236> [CR](#)

[BibTeX info](#)

Communicated by Jean-Michel Morel, Miguel Colom

Demo edited by Jose-Luis Lisani

Abstract

In this paper we describe the implementation of the algorithm for local contrast enhancement published by Caselles et al. in 1999. This algorithm was the first designed explicitly to increase the contrast while preserving the so-called 'shape structure' of the image, that is, its set of level sets. According to the mathematical morphology school, artifacts are created when this structure is modified. The original algorithm is described and also two alternative implementations are proposed, which limit the over-enhancement of noise.

Download

- full text manuscript: [PDF low-res. \(957KB\)](#) [PDF \(92.7MB\) ?](#)
- source code: [TAR/GZ](#) [SWHID info](#) [</>](#)



“IPOL is a research journal of image processing and image analysis which emphasizes the role of mathematics as a source for algorithm design and the reproducibility of the research.”
<http://www.ipol.im/>