

Déposer le code source d'un logiciel dans HAL

Bonnes pratiques et guide utilisateur

Version 2.0 (mars 2022)

Auteurs :

- Morane Gruenpeter morane@softwareheritage.org
- Jozefina Sadowska jozefina.sadowska@inria.fr
- Estelle Nivault estelle.nivault@inria.fr
- Alain Monteil alain.monteil@inria.fr



HAL
open science

CCSD ●●●

Inria



Software Heritage

Table de matières: Dépôt logiciel sur HAL

[Introduction](#)

[Checklist](#)

[Chapitre 1: Préparer le code source pour archivage](#)

Dépôt local: .zip /.tar.gz

[Chapitre 2: Dépôt des fichiers sources](#)

Dépôt SWHID: with the SoftWare Heritage ID

[Chapitre 3: Dépôt des métadonnées avec SWHID](#)

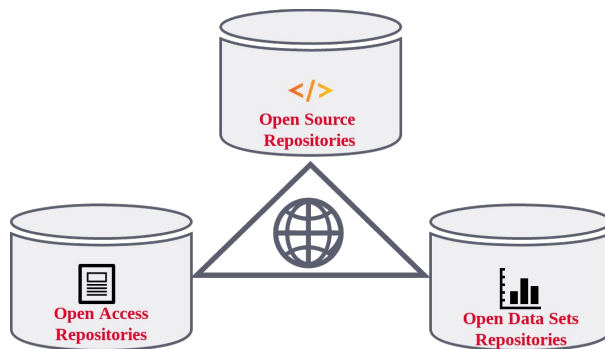
[Chapitre 4: Le processus de modération](#)

[Chapitre 5: La publication et les exports](#)

[Références](#)

Introduction: Pourquoi déposer sur HAL?

- ★ **Archiver** le logiciel sur HAL et sur SWH
 - Grande **visibilité** aux logiciels dans une démarche de science ouverte
 - **Archivage pérenne**, en transférant votre code vers Software Heritage, l'archive universelle du code source
- ★ **Identifier**
 - les objets avec un **SWHID** (SoftWare Heritage Identifier)
 - la notice et la citation avec un **HAL-ID**
- ★ **Décrire** le logiciel avec des métadonnées qui sont vérifiées
 - **Modération** des métadonnées par l'équipe des documentalistes
- ★ **Citer** le dépôt avec une citation complète
 - Différents **formats d'export** pour faciliter la citation



Trois piliers de la Science Ouverte, Software Heritage CC-BY 4.0 2019

Introduction: Quel objet logiciel déposer sur HAL?

- ★ Le **code source** d'un logiciel (ne pas déposer les exécutables)
- ★ Le logiciel qui est développé dans un **milieu académique**
- ★ Seuls **les créateurs/auteurs** du logiciel ou leurs représentants peuvent déposer un logiciel sur HAL

Cas d'utilisations

Je **développe** mes logiciels **localement** et je les partage sur mon **site web** personnel ou institutionnel

Dépôt source: déposer .zip /.tar.gz

Je **développe** mes logiciels sur une **plateforme collaborative** (type GitHub, GitLab, Bitbucket...)

Dépôt SWHID: déposer métadonnées et SWHID

Introduction: Les étapes du dépôt logiciel

- Dépôt sur HAL
 - ◆ dépôt source ou dépôt SWHID
- Validation du formulaire par le contributeur
- Dépôt en cours de **modération**
 - ◆ **Dialogue** entre contributeur et modérateur
- **Validation** du dépôt en modération
- Dépôt **publié sur HAL** et **transfert vers SWH**
- Formats d'exports
 - ◆ Citation
 - ◆ BibTeX
 - ◆ Codemeta.json
 - ◆ TEI

Checklist par cas d'utilisation

Dépôt source: déposer .zip / .tag.gz

- 1. Préparer le logiciel (localement)**
 - Fichiers AUTHORS, LICENSE & README
 - Compresser les documents en .zip / .tar.gz
- 2. Déposer l'archive compressée**
- 3. Compléter les métadonnées**
 - Choisir le type de dépôt
 - Ajouter des métadonnées générales
 - Ajouter des métadonnées spécifiques aux logiciels
 - Renseigner les auteurs
 - Valider le dépôt

Dépôt SWHID: déposer métadonnées et SWHID

- 1. Préparer le logiciel (sur un code repository)**
 - Fichier AUTHORS, LICENSE & README
 - Fichier codemeta.json
 - [Save code now](#) sur Software Heritage
 - Choisir un SoftWare Heritage IDentifier (SWHID)
- 2. Déposer le lien (SWHID) vers le contenu**
- 3. Compléter les métadonnées**
 - Ajouter le domaine
 - Contrôler les entrées codemeta
 - Compléter des métadonnées
 - Renseigner les auteurs
 - Valider le dépôt

Chapitre 1:

Préparer le code source pour archivage

1.1 Préparer le logiciel

1.1.1 Ajouter les fichiers suivants :

Ces *éléments*
sont *vérifiés* par
les
modérateurs

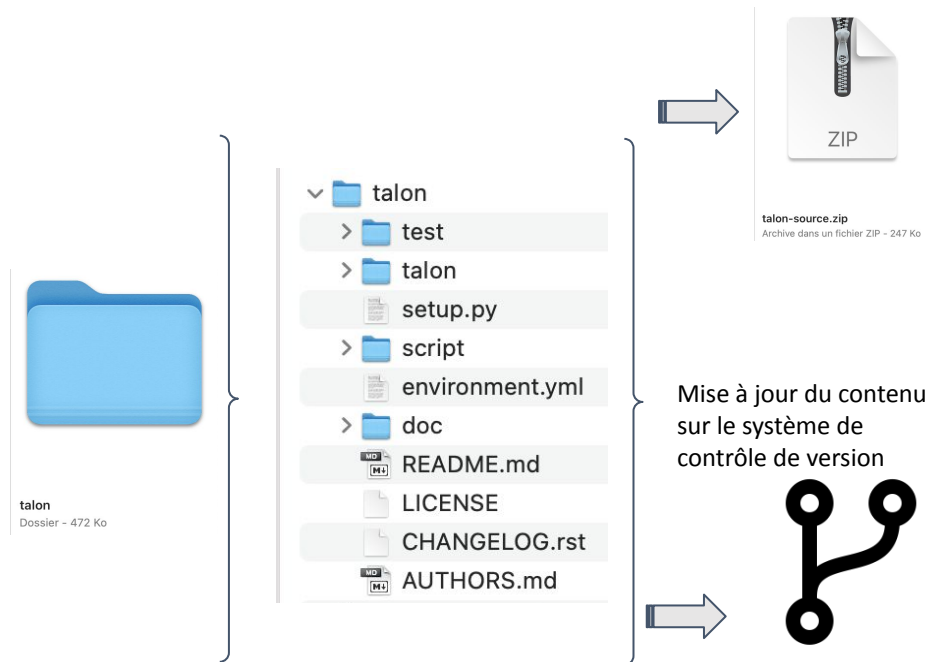
- README
- AUTHORS
- LICENSE (à choisir avec les titulaires des droits patrimoniaux du dit logiciel - [liste de référence SPDX](#))

1.1.2 Dépôt source: créer une archive compressée .zip ou .tar.gz

Nommer de préférence avec le nom du logiciel et sa version

1.1.3 Dépôt SWHID: utiliser votre système de contrôle de version

Dépôt source: déposer .zip / .tar.gz



Dépôt SWHID : déposer métadonnées et SWHID

1.2 Fichier README

★ Obligatoire:

- Nom du logiciel / projet
- Description du projet

★ Recommandé:

- Site du projet
- lien vers la documentation
- Contact & support
- Liste des fonctionnalités
- Environnement de développement
 - build, installation, requirements
 - comment exécuter le code

★ Possible:

- Usage - comment utiliser le code
- Nouvelles récentes du projet
- Visuels du logiciel

```
This is Python version 3.8.0 alpha 0
```

```
-----  
.. image:: https://travis-ci.org/python/cpython.svg?branch=master  
:alt: CPython build status on Travis CI  
:target: https://travis-ci.org/python/cpython  
  
.. image:: https://ci.appveyor.com/api/projects/status/4mew1a93xdkbf5ua/branch/master?svg=true  
:alt: CPython build status on Appveyor  
:target: https://ci.appveyor.com/project/python/cpython/branch/master  
  
.. image:: https://dev.azure.com/python/cpython/_apis/build/status/Azure%20Pipelines%20CI?branchName=master  
:alt: CPython build status on Azure DevOps  
:target: https://dev.azure.com/python/cpython/_build/latest?definitionId=4&branchName=master  
  
.. image:: https://codecov.io/gh/python/cpython/branch/master/graph/badge.svg  
:alt: CPython code coverage on Codecov  
:target: https://codecov.io/gh/python/cpython  
  
.. image:: https://img.shields.io/badge/zulip-join_chat-brightgreen.svg  
:alt: Python Zulip chat  
:target: https://python.zulipchat.com
```

```
Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011,  
2012, 2013, 2014, 2015, 2016, 2017, 2018 Python Software Foundation. All rights  
reserved.
```

```
See the end of this file for further copyright and license information.
```

```
.. contents::
```

```
General Information
```

```
-----
```

```
- Website: https://www.python.org  
- Source code: https://github.com/python/cpython  
- Issue tracker: https://bugs.python.org  
- Documentation: https://docs.python.org  
- Developer's Guide: https://devguide.python.org/
```

```
Contributing to CPython
```

```
-----
```

```
For more complete instructions on contributing to CPython development,  
see the `Developer Guide`_.
```

```
.. _Developer Guide: https://devguide.python.org/
```

```
Using Python
```

```
-----
```

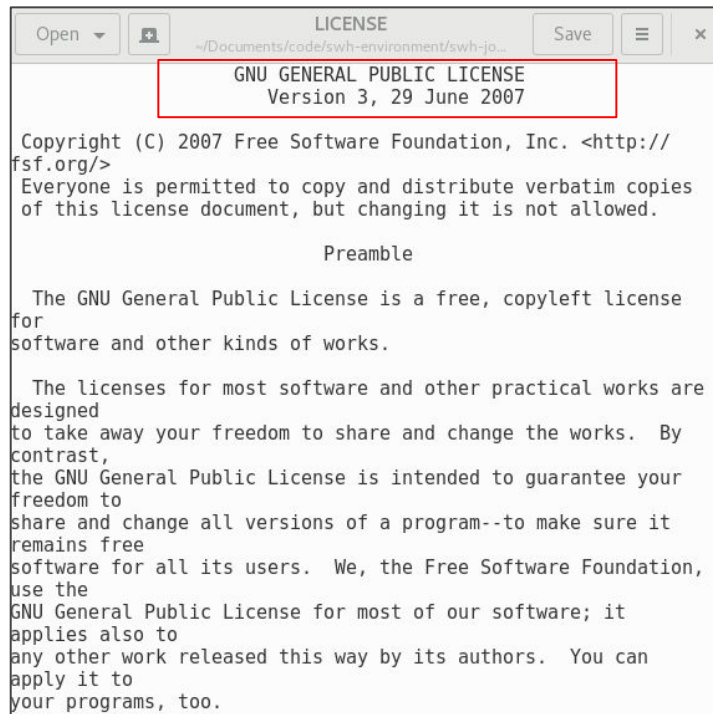
```
Installable Python kits, and information about using Python, are available at  
`python.org`_.
```

Readme du projet CPython archivé sur SWH :

swh:1.cnt:970f62e6938908a9caaaa0a07fe425bd3976c101:origin=https://github.com/python/cpython/

1.4 Fichier LICENSE

- ★ Avant tout dépôt de code source, vous devez réfléchir à la licence que vous souhaitez pour votre code.
 - Chez INRIA, votre CPPI est votre interlocuteur pour discuter de ces questions.
- ★ Consulter les personnes en charge dans votre établissement
- ★ Si vous avez plus d'une licence, créez un répertoire LICENSES/ avec toutes les licences
- ★ Noter la/les licence/s utilisé/s dans le code source dans les métadonnées
 - La cohérence entre la licence dans le formulaire sur HAL et dans le code est vérifié par les modérateurs
 - Le **déposant est responsable** de la compatibilité entre licences (entre le code déposé et ses dépendances)



```
LICENSE
~/Documents/code/sw-h-environment/sw-h-jo...
GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://
fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

                Preamble

The GNU General Public License is a free, copyleft license
for
software and other kinds of works.

The licenses for most software and other practical works are
designed
to take away your freedom to share and change the works. By
contrast,
the GNU General Public License is intended to guarantee your
freedom to
share and change all versions of a program--to make sure it
remains free
software for all its users. We, the Free Software Foundation,
use the
GNU General Public License for most of our software; it
applies also to
any other work released this way by its authors. You can
apply it to
your programs, too.
```

Ressources pour examiner différentes licences:

- <https://choosealicense.com/>
- <https://reuse.software/>

Chapitre 2:

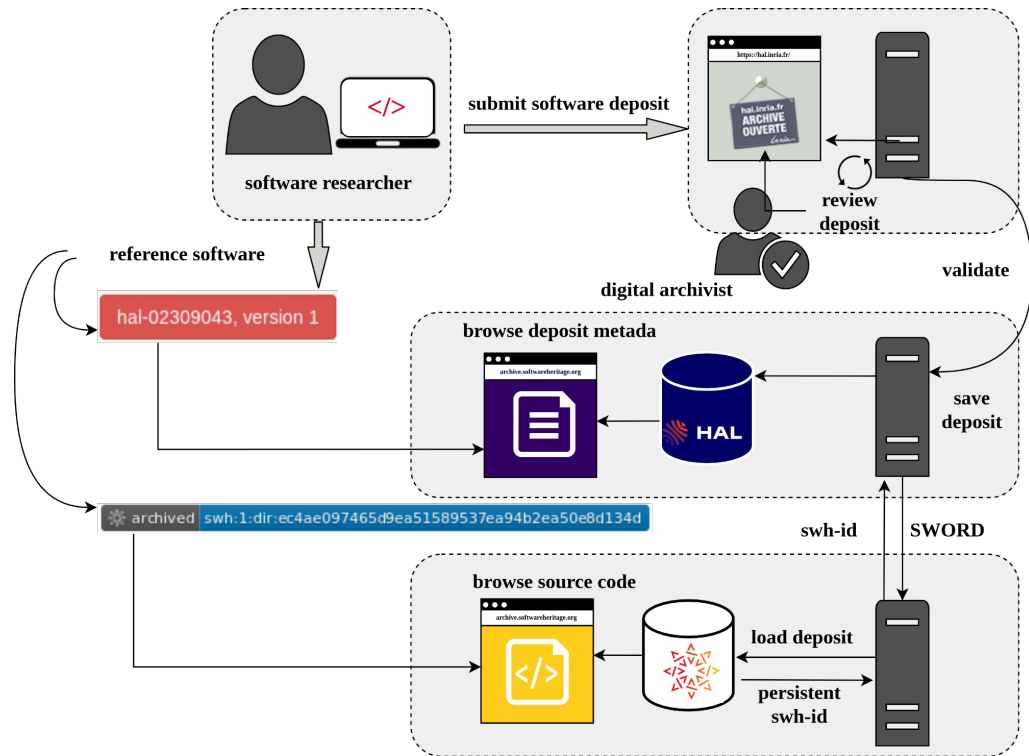
Dépôt des fichiers sources

Dépôt source: déposer .zip /.tar.gz

Dépôt source: déposer .zip /.tar.gz

Le dépôt source

- une **archive** compressée contenant du code source (majoritairement des fichiers textes)
- une collection de **métadonnées**
 - ◆ métadonnées générales
 - ◆ métadonnées spécifiques



2.1: Déposer les fichiers

Depuis l'onglet "Déposer", sélectionnez l'archive compressée ou Glissez/Déposez

Il n'est pas possible d'ajouter plusieurs archives ou plusieurs fichiers distincts.

Le .zip doit contenir tous les fichiers et ne doit pas être décompressé avant la validation.

Le .zip ne doit pas contenir un autre .zip



The screenshot shows the HAL website's user interface. At the top, there is a dark blue header with the HAL logo and the text "science ouverte". To the right, it says "La connaissance libre et partagée". Below the header is a navigation menu with links: Accueil, Dépôt, Consultation, Recherche, Documentation, and Mon espace. The main content area is titled "Déposer le(s) fichier(s)". It contains a large dashed box for file upload with a cloud icon and a document icon, and the text ".zip / .tar.gz" and "Glissez-déposez ou cliquez pour choisir vos fichiers". Below this, it says "Taille maximale du fichier : 200M". To the right of the upload area is a section titled "Chargez les métadonnées à partir d'un identifiant" with a text input field containing "swih:1.dir:79b8c8755dbed3401a6a7184ffc196f3c58cb5d.origi" and a button "Récupérer les métadonnées". At the bottom, there is a small note: "Si vous souhaitez ajouter un embargo, récupérer des fichiers de votre espace FTP ou toute autre action avancée, veuillez afficher la vue détaillée."

2.2: Compléter les métadonnées

→ 2.2.1 Choisir le type de document* :

Logiciel

Pour afficher la liste complète des métadonnées cocher la case

*Les métadonnées **obligatoires** sont indiqués par un **

→ 2.2.2 Ajouter les métadonnées

générales :

- Nom*
- Domaine*
- Description (*Il est recommandé d'avoir une description cohérente avec le contenu du fichier README*)
- Mots-clés
- Identifiants
- Données associées
- Publications associées
- Date de production/écriture
- Editeur
- Institution, projets (ANR/ERC), etc.

[Afficher la liste complète des métadonnées](#)

Avant tout dépôt de code source, vous devez réfléchir à la licence que vous souhaitez pour votre code. Chez INRIA, votre CPPI est votre interlocuteur pour discuter de ces questions.

* Champs requis

Type de document *
Logiciel

Nom *
Vous pouvez renseigner le titre en plusieurs langues : choisir la langue et cliquer sur + pour ajouter une nouvelle langue
TALON: Tractograms As Linear Operators in Neuroimaging
anglais +







français +

Domaine *
Le domaine que vous mettez en premier, sera considéré comme la discipline principale du dépôt (article, logiciel ou autre). L'ordre des domaines peut être changé par glissé/déposé.
+ Imagerie médicale
+ Traitement du signal et de l'image [eess.SP]
Ajouter un domaine

Description
Cliquer sur + pour ajouter une nouvelle langue
TALON is a pure Python package that implements Tractograms As Linear Operators in Neuroimaging. The software provides the TALON Python module, which includes all the functions and tools that are necessary for filtering a tractogram. In particular, specific functions are devoted to:
- Transforming a tractogram into a linear operator.
- Solving the inverse problem associated to the filtering of a tractogram.
anglais +

→ 2.2.3 Ajouter les **métadonnées spécifiques** :

- Licences* (s'appuie sur le [référentiel SPDX](#), il est aussi possible de saisir une licence qui ne figure pas dans cette liste)
- Langage de programmation
- Code repository
- Platform/OS
- Version
- Etat du développement
- Outils de développement

Licences *	<i>La ou les licences sous lesquelles est publié ce logiciel (vous pouvez utiliser l'autocomplétion)</i>
	<input type="text" value="MIT License"/> 
	<input type="text"/> 
Langage de programmation	<input type="text" value="Python"/> 
	<input type="text"/> 
Code Repository	<i>Lien où se trouve le développement du code (SVN, github, gitlab, CodePlex).</i>
	<input type="text" value="https://gitlab.inria.fr/cobcom/talon"/>
Platform/OS	<i>Le système d'exploitation compatible avec le logiciel</i>
	<input type="text"/> 
Version	<i>la version du logiciel (peut être différente de la version publiée sur HAL)</i>
	<input type="text" value="0.3.0"/>
Etat du développement	<i>L'état du développement du logiciel au moment du dépôt (Concept, WIP, Suspendu, Actif, Inactif)</i>
	<input type="text" value="active"/>
Outils de développement	<i>Les outils de développement associés au logiciel (Framework, middleware, plateforme logicielle)</i>
	<input type="text"/> 

2.3 Compléter les données **auteur(s)**

- Ajouter un auteur
- Ajouter une affiliation (*au moins un auteur doit être affilié*)

- Il est possible d'ajouter plusieurs auteurs et de leur associer une fonction (développement, maintenance, design, architecture, débogage, documentation, test, support, management).

- Il est nécessaire de renseigner le/les auteurs qui sont mentionnés dans le fichier AUTHORS.

- Vous pouvez ajouter un fichier CONTRIBUTORS pour les personnes qui ne sont pas auteurs dans la notice.

The screenshot displays a web interface titled "Compléter les données auteur(s)" with a green checkmark. It lists four authors, each with a list of affiliations and a dropdown menu for selecting a function. The authors are:

- Matteo Frigo (IdHAL: matteofrigo) - Auteur
 - ATHENA - Computational Imaging of the Central Nervous System
 - UCA - Université Côte d'Azur
 - Ajouter une affiliation
- Mauro Zucchelli (IdHAL: mauro-zucchelli) - Auteur
 - ATHENA - Computational Imaging of the Central Nervous System
 - UCA - Université Côte d'Azur
 - Ajouter une affiliation
- Rachid Deriche - Auteur
 - ATHENA - Computational Imaging of the Central Nervous System
 - UCA - Université Côte d'Azur
 - Ajouter une affiliation
- Samuel Deslauriers-Gauthier - Auteur
 - ATHENA - Computational Imaging of the Central Nervous System
 - UCA - Université Côte d'Azur


A dropdown menu for Matteo Frigo is open, showing a list of functions: Développement (checked), Maintenance, Design, Architecture, Débogage, Documentation, Test, Support, and Management. The option "Choisir la fonction" is highlighted with a red box. Other options visible in the interface include "Afficher toutes les options", "Pour cet auteur", and "Pour cette structure".

2.4 Valider le dépôt

- Valider le transfert et contribuer au patrimoine logiciel constitué par Software Heritage
- Accepter les conditions
- Cliquer sur Déposer

☰ Valider le dépôt ✓

Logiciel
Matteo Frigo, Mauro Zucchelli, Rachid Deriche, Samuel Deslauriers-Gauthier. TALON: Tractograms As Linear Operators in Neuroimaging. 2021

 Software Heritage Voir les conditions pour le transfert

Software Heritage a pour objectif de collecter, préserver, et rendre accessible, à tous, le code source de tous les logiciels disponibles.

Pour pouvoir transférer votre logiciel dans la plus grande archive de code source, votre dépôt doit satisfaire les conditions suivantes :

- Les fichiers déposés doivent être sous une licence libre.
- Les fichiers ne peuvent pas être sous embargo.

L'accès à votre dépôt sur Software Heritage sera disponible dans un délai de 7-30 jours (le temps de traitement de votre dépôt)

Conditions

En déposant ce document, le contributeur (je) accorde la licence suivante à HAL :

- J'autorise HAL à mettre en ligne et à distribuer cet article ;
- Je reconnais avoir pris connaissance que les dépôts ne peuvent pas être supprimés, une fois acceptés ;
- Je comprends que HAL se réserve le droit de reclasser ou de rejeter tout dépôt.

J'accepte ces conditions

Vider Annuler **↓ Déposer**

Chapitre 3:

Dépôt des métadonnées avec SWHID

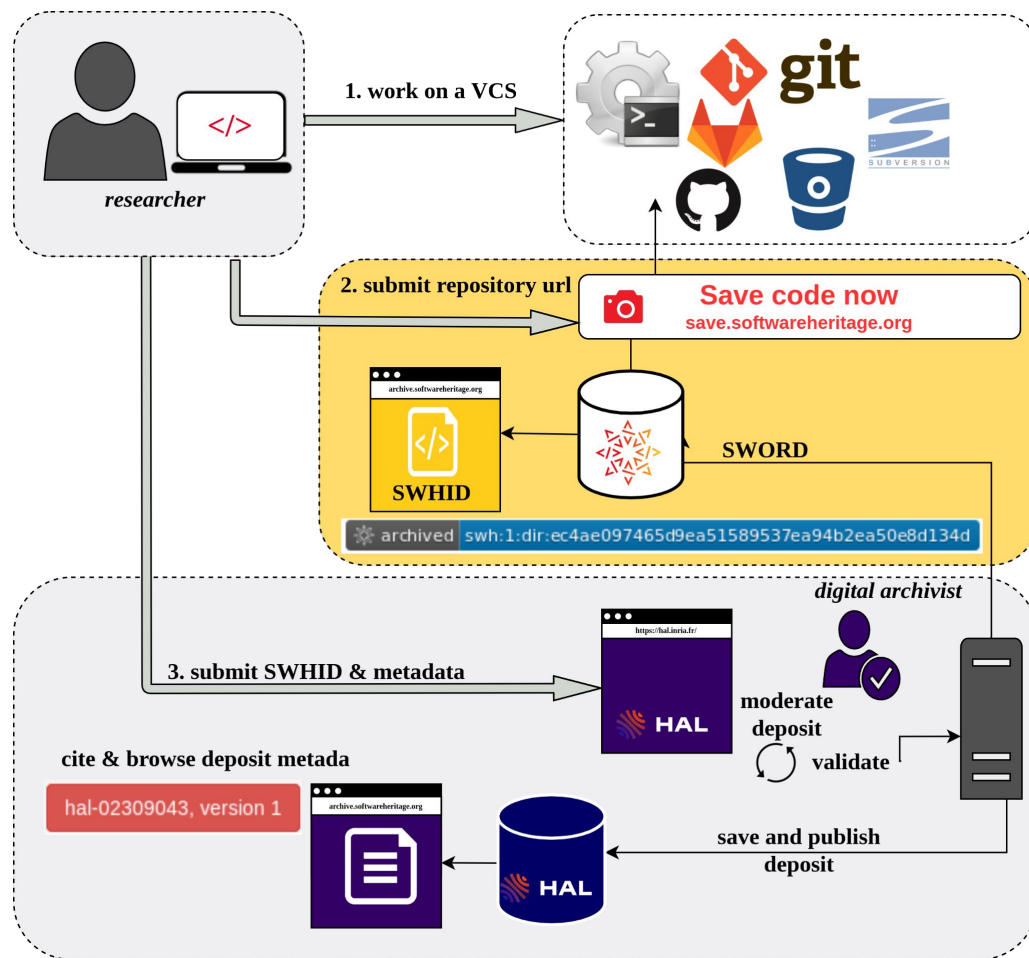
Dépôt SWHID : déposer métadonnées et SWHID

Dépôt SWHID : déposer métadonnées et SWHID

Un dépôt SWHID

Le dépôt SWHID peut être fait avec:

- un **SWHID** fourni par Software Heritage. Celui-ci référence le code source archivé.
- une collection de **métadonnées**
 - ◆ **Charger** les métadonnées à partir d'un **SWHID**. Les informations sur un fichier **codemeta.json** permettront de remplir automatiquement votre dépôt.
 - ◆ **Remplir** le formulaire avec des métadonnées.



*VCS = Version Control System

3.1 Préparation du logiciel

Avant le dépôt sur HAL, préparez le code source du logiciel sur la plateforme de développement (GitHub, Gitlab, etc.).

Ces éléments sont vérifiés après par les modérateurs.

3.1.1 ajoutez les fichiers suivants sur le `code repository` :

- ❑ README
- ❑ AUTHORS
- ❑ LICENSE (à choisir avec les titulaires des droits patrimoniaux du dit logiciel - [liste de référence](#))

- ❑ **codemeta.json** - facultatif mais pratique car les métadonnées sont ajoutées dans le formulaire HAL automatiquement en utilisant le SWHID approprié

The screenshot shows a GitHub repository page for 'moranegg/AffectationRO'. The file list includes 'bin', 'glpk_files', 'src', 'AUTHORS', 'LICENSE', 'README.md', and 'codemeta.json'. The 'README.md' file is selected, showing the title 'projet en Recherche Opérationnelle' and a DOI '10.5281/zenodo.438684'. The content of the README describes a distributed system with parallel processes.

File	Description	Age
bin	Control class added (GestionSolution) and changed packages	6 years ago
glpk_files	entity problem added	6 years ago
src	Control class added (GestionSolution) and changed packages	6 years ago
AUTHORS	add AUTHORS file	4 years ago
LICENSE	Create LICENSE	4 years ago
README.md	Update Readme with DOI	4 years ago
codemeta.json	Add CodeMeta file for metadata	2 months ago

projet en Recherche Opérationnelle

DOI: 10.5281/zenodo.438684

Problème d'affectation généralisée

un système distribué comme un ensemble de processeurs pouvant exécuter des tâches (ou processus) en parallèle. On considère donc un ensemble de m processeurs, chacun muni d'une certaine quantité de mémoire vive (RAM), qu'il peut utiliser pour charger et exécuter des tâches, et un ensemble de n tâches à exécuter, chacune nécessitant une certaine quantité de RAM pour être chargée et exécutée. Cette quantité peut en fait varier en fonction de la nature du processeur sur lequel la tâche est exécutée, et dépend donc du choix de ce processeur. Enfin, à chaque couple (processeur, tâche), on associe un coût à payer pour exécuter cette tâche sur ce processeur, et à chaque couple de tâches on associe un coût de communication (coût à payer si ces deux tâches sont exécutées sur des processeurs différents).

3.1.2 Pourquoi CodeMeta ?

- Un vocabulaire qui étend le vocabulaire schema.org
 - <https://codemeta.github.io/terms/>
- Une [communauté académique](#)
- Une [table de correspondances](#) qui permet de traduire plusieurs ontologies/vocabulaires vers CodeMeta

Outil pour créer un codemeta.json

CodeMeta generator

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

Pour créer facilement un fichier codemeta.json utiliser l'[outil en ligne](#)



[Pour contribuer](#)

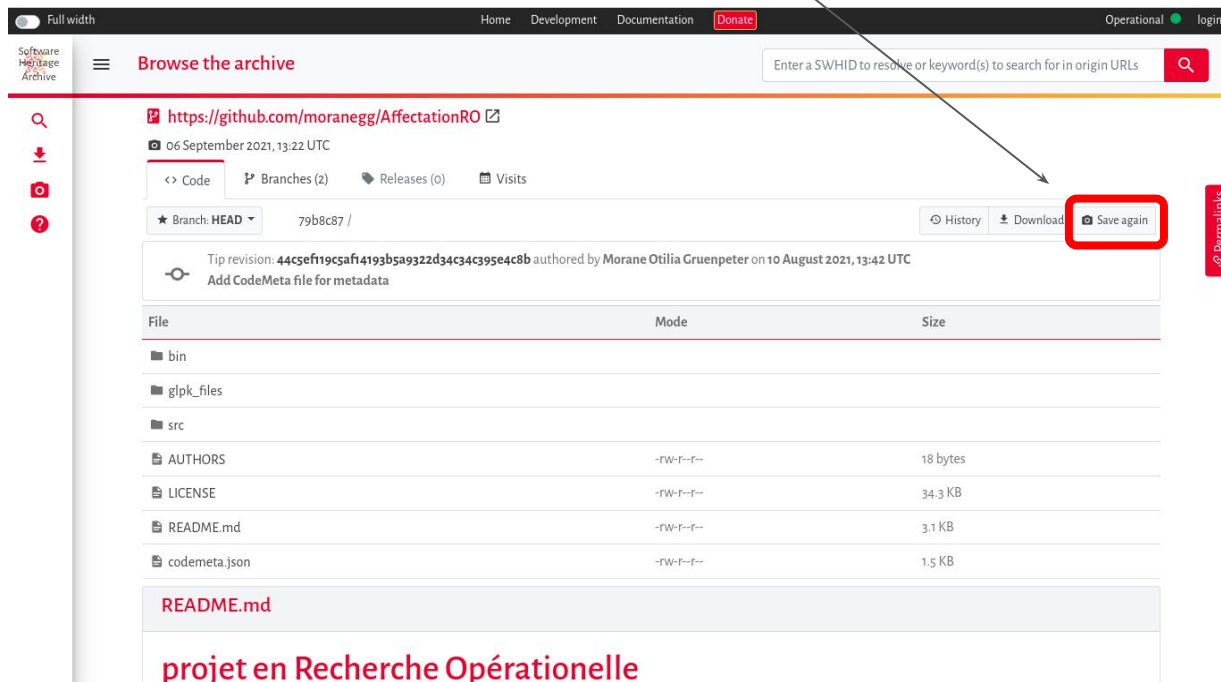
3.1.3 Fichier CodeMeta: un exemple

```
{
  "@context": "https://doi.org/10.5063/schema/codemeta-2.0",
  "@type": "SoftwareSourceCode",
  "license": "https://spdx.org/licenses/AGPL-3.0",
  "codeRepository": "https://github.com/moranegg/AffectationRO",
  "dateCreated": "2014-01-01",
  "dateModified": "2019-06-26",
  "name": "AffectationRO- The assignment problem",
  "version": "1.0.0",
  "description": "A java implementation for The Assignment Problem a distributed system as a set of processors that can perform tasks (or processes) in parallel. We therefore consider a set of m processors, each equipped with a certain amount of random access memory (RAM).\r\n\r\n We associate a cost to pay to perform this task on this processor, and each pair of tasks is associated with a communication cost. The Assignment problem works on minimizing the cost and maximizing the tasks performed.",
  "applicationCategory": "info",
  "releaseNotes": "First release with GLPK, in Beta testing",
  "developmentStatus": "concept",
  "keywords": [
    "distributed systems",
    "glpk",
    "optimisation",
    "OR"
  ],
  "programmingLanguage": [
    "Java"
  ],
  "author": [
    {
      "@type": "Person",
      "@id": "https://orcid.org/0000-0002-9777-5560",
      "givenName": "Morane",
      "familyName": "Gruenpeter",
      "email": "morane.gg@gmail.com",
      "affiliation": {
        "@type": "Organization",
        "name": "Software Heritage"
      }
    }
  ]
}
```

3.2 Sauvegarder le code sur Software Heritage

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

3.2.1 Vérifier si l'archive est sur SWH et la mettre à jour



Full width Home Development Documentation Donate Operational login

Software Heritage Archive

Browse the archive

Enter a SWHID to resolve or keyword(s) to search for in origin URLs

<https://github.com/moranegg/AffectationRO>

06 September 2021, 13:22 UTC

<> Code Branches (2) Releases (0) Visits

★ Branch: HEAD 79b8c87 / History Download Save again

Tip revision: [44c5ef19c5af14193b5a9322d34c34c395e4c8b](#) authored by Morane Otília Gruenpeter on 10 August 2021, 13:42 UTC
Add CodeMeta file for metadata

File	Mode	Size
bin		
glpk_files		
src		
AUTHORS	-rw-r--r--	18 bytes
LICENSE	-rw-r--r--	34.3 KB
README.md	-rw-r--r--	3.1 KB
codemeta.json	-rw-r--r--	1.5 KB

README.md

projet en Recherche Opérationnelle

Permalinks

3.2.2 Si le code n'est pas sur Software Heritage - **Soumettre** le code repository avec le **Save Code Now**:

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

The screenshot shows the 'Save code now' interface on the Software Heritage website. The page has a header with the Software Heritage logo and a search bar. A left sidebar lists features: Search, Downloads, Save code now (highlighted), and Help. The main content area is titled 'Save code now' and contains a form with two fields: 'Origin type' and 'Origin url'. The 'Origin type' field is a dropdown menu with 'git' selected. The 'Origin url' field is an empty text input. A 'Submit' button is located to the right of the 'Origin url' field. Three numbered callouts are overlaid on the form: 1. A purple circle with the number '1' points to the 'git' dropdown. 2. A purple circle with the number '2' points to the 'Origin url' input field. 3. A purple circle with the number '3' points to the 'Submit' button. Three grey callout boxes with black borders contain the following text: '1. Choisir le type de système de contrôle de version' (pointing to callout 1), '2. Ajouter l'url' (pointing to callout 2), and '3. Soumettre' (pointing to callout 3).

Software Heritage Archive

Save code now

Enter a SWHID to resolve or keyword(s) to search for in origin URLs

You can contribute to extend the content of the Software Heritage archive by submitting code repositories. To do so, fill the required info in the form below:

Origin type Origin url

git Submit

1. Choisir le type de système de contrôle de version

2. Ajouter l'url

3. Soumettre

3.3 Choisir un SWHID sur l'archive Software Heritage

1. Cliquer sur `Permalinks`

2. Choisir le type d'objet - `directory`

3. Ajouter le contexte du SWHID

4. Copier l'identifiant sur le formulaire HAL

Full width Home Development Documentation Donate Operational login

Software Heritage Archive

Browse the archive

Enter a SWHID to resolve or keyword(s) to search for in origin URLs

https://github.com/moranegg/AffectationRO

06 September

<> Code Branches Releases (0) Visits

Branch: HEAD 79

Tip revision: 44c5ef119c5af14193b5a9322d34c34c395e4c8b

Add CodeMeta file for repository

File

- bin
- glpk_files
- src
- AUTHORS
- LICENSE
- README.md
- codemeta.json

Permalinks

To refer to this repository in the current URI scheme will remain the same over time).

Select the type of object to archive

directory revision snapshot

archived repository archived swh:1:dir:79b8c8755dbed34f01a6a7184ffc196f3c58cb5d

Iframe embedding

```
swh:1:dir:79b8c8755dbed34f01a6a7184ffc196f3c58cb5d;
origin=https://github.com/moranegg/AffectationRO;
visit=swh:1:snp:3d73c454047658280dbbf7a5fbc3c25f788d4a37;
anchor=swh:1:rev:44c5ef119c5af14193b5a9322d34c34c395e4c8b
```

Add contextual information

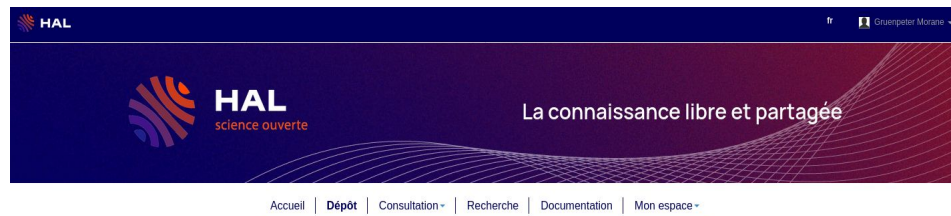
Copy identifier Copy permalink

README.md

projet en Recherche Opérationnelle

3.4 Déposer SWHID - la référence vers le contenu

- Mettre le **SWHID** sur le formulaire HAL
 - ◆ préférer un SWHID avec contexte (pour garder le lien vers l'historique de développement)
 - ◆ Si un codemeta.json existe dans le code source - la plateforme HAL pourra récupérer les métadonnées automatiquement
- Vérifier et compléter les métadonnées
 - ◆ vérifier les métadonnées insérées
 - ◆ compléter les métadonnées manquantes
 - ◆ choisir le domaine
 - ◆ vérifier les auteurs et affiliations
- Valider le dépôt



Vous pouvez déposer un fichier et/ou compléter votre dépôt à partir d'un identifiant externe.

Chargez les métadonnées à partir d'un identifiant

Les informations associées à cet identifiant permettront de remplir automatiquement votre dépôt. Par exemple, le DOI est un code qui apparaît généralement sur la page de votre article sur le site de l'éditeur.

SWHID sw/h:1.dlr:79b8c8755d8ed34401a6a7184ffc1963c58cb5d.origi

Récupérer les métadonnées

Si vous souhaitez ajouter un embargo, récupérer des fichiers de votre espace FTP ou toute autre action avancée, veuillez afficher la vue détaillée.

SWHID

Chapitre 4:

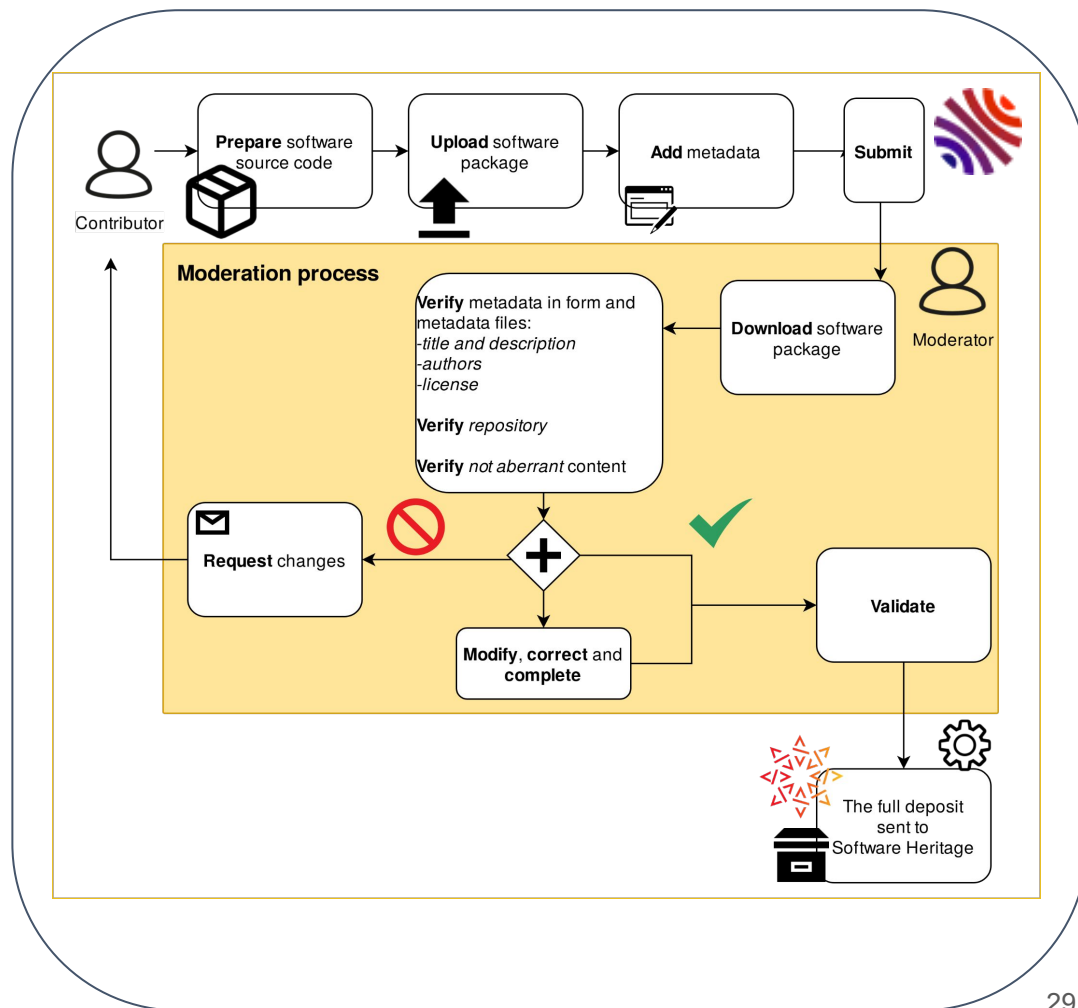
Le processus de modération

Le processus de modération

- Dépôt en cours de **modération**
- **Dialogue** entre contributeur et modérateur
- **Validation** du dépôt en modération

Guide de bonnes pratiques :

Morane Gruenpeter, Jozefina Sadowska. La modération d'un dépôt logiciel : Bonnes pratiques et guide pour le modérateur. [Rapport Technique] Inria; CCSD; Software Heritage. 2018. [hal-01876705](https://hal.archives-ouvertes.fr/hal-01876705)



Chapitre 5:

La publication et les exports

5.1 Le dépôt en cours de transfert vers SWH

→ Le dépôt source est transféré vers SWH:

- ◆ Contenu
- ◆ Métadonnées

→ Le dépôt SWHID est transféré vers SWH:

- ◆ Métadonnées

Dépôt sources: Pendant cette première phase c'est le dossier zippé avec le code déposé qui est visible depuis Hal - pas de lien avec SWH

hal-02522751, version 1

TALON: Tractograms As Linear Operators in Neuroimaging

Matteo Frigo^{1,2}, Mauro Zucchelli^{1,2}, Rachid Deriche^{1,2}, Samuel Deslauriers-Gauthier^{1,2} [Détails](#)

- 1 ATHENA - Computational Imaging of the Central Nervous System
CRISAM - Inria Sophia Antipolis - Méditerranée
- 2 UCA - Université Côte d'Azur

Abstract : TALON is a pure Python package that implements Tractograms As Linear Operators in Neuroimaging. The software provides the TALON Python module, which includes all the functions and tools that are necessary for filtering a tractogram. In particular, specific functions are devoted to: - Transforming a tractogram into a linear operator. - Solving the inverse problem associated to the filtering of a tractogram.

Type de document : [Logiciel](#)

Domaine : [Informatique \[cs\]](#) / [Imagerie médicale](#)
[Informatique \[cs\]](#) / [Traitement du signal et de l'image \[eess.SP\]](#)

Liste complète des métadonnées [Voir](#)

TÉLÉCHARGER

Fichier : talon-source.zip [Télécharger](#)

<https://hal.halpreprod.archives-ouvertes.fr/hal-02522751>
Contributeur : [Estelle Nivault](#) [Contacter le contributeur](#)
Soumis le : vendredi 28 mai 2021 - 16:02:01
Dernière modification le : lundi 31 mai 2021 - 18:08:40

MÉTADONNÉES

Keywords : [Diffusion MRI](#) [dMRI](#) [tractography](#) [python](#)
[optimization](#)

version [0.3.0](#)

Licences [MIT License](#)

Langage de programmation [Python](#)

Code Repository <https://gitlab.inria.fr/cobcom/talon>

Outils de développement [Active](#)

CITATION

Matteo Frigo, Mauro Zucchelli, Rachid Deriche, Samuel Deslauriers-Gauthier. TALON: Tractograms As Linear Operators in Neuroimaging. 2021. (hal-02522751)

EXPORTER

[CodeMeta](#) [BibTeX](#) [TEI](#) [DC](#) [DCterms](#)

[EndNote](#)

PARTAGER

[f](#) [t](#) [e](#) [+](#)

5.2 Le dépôt final

hal-02522751, version 1

TALON: Tractograms As Linear Operators in Neuroimaging

Matteo Frigo^{1,2}, Mauro Zucchelli^{1,2}, Rachid Deriche^{1,2}, Samuel Deslauriers-Gauthier^{1,2} [Détails](#)

1 ATHENA - Computational Imaging of the Central Nervous System

CRISAM - Inria Sophia Antipolis - Méditerranée

2 UCA - Université Côte d'Azur

Abstract : TALON is a pure Python package that implements Tractograms As Linear Operators in Neuroimaging. The software provides the TALON Python module, which includes all the functions and tools that are necessary for filtering a tractogram. In particular, specific functions are devoted to: - Transforming a tractogram into a linear operator. - Solving the inverse problem associated to the filtering of a tractogram.

Type de document : [Logiciel](#)

Domaine :

Informatique [cs] / Imagerie médicale

Informatique [cs] / Traitement du signal et de l'image [eess.SP]

Liste complète des métadonnées [Voir](#)

CONSULTER



Software Heritage

swh:1:dir:f25157ad1b13cb20ac3457d4f6756b49ac63d079;origin=https://inria.halpreprod.archives-ouvertes.fr/hal-02522751;visit=swh:1:snp:8a2cb6ecd1478c63550e524a5723e06597259a07;anchor=swh:1:rev:5c9642f43d76c7f1c22e1bb641561e210eb52a94;path=/

[Consulter](#)

<https://hal.halpreprod.archives-ouvertes.fr/hal-02522751>

Contributeur : Estelle Nivault Connectez-vous pour contacter le contributeur

Soumis le : vendredi 28 mai 2021 - 16:02:01

Dernière modification le : mercredi 2 juin 2021 - 13:59:59

MÉTADONNÉES

Keywords : [Diffusion MRI](#) [dMRI](#) [tractography](#) [python optimization](#)

version

[0.3.0](#)

Licences

[MIT License](#)

Langage de programmation

[Python](#)

Code Repository

<https://gitlab.inria.fr/cobcom/talon>

Outils de développement

[Active](#)

CITATION

Matteo Frigo, Mauro Zucchelli, Rachid Deriche, Samuel Deslauriers-Gauthier. TALON: Tractograms As Linear Operators in Neuroimaging. 2021. (swh:1:dir:f25157ad1b13cb20ac3457d4f6756b49ac63d079;origin=https://inria.halpreprod.archives-ouvertes.fr/hal-02522751;visit=swh:1:snp:8a2cb6ecd1478c63550e524a5723e06597259a07;anchor=swh:1:rev:5c9642f43d76c7f1c22e1bb641561e210eb52a94;path=/). (hal-02522751)

EXPORTER

[CodeMeta](#) [BibTeX](#) [TEI](#) [DC](#) [DCterms](#)

[EndNote](#)

Pour consulter le contenu directement sur SWH

Citation et exports

5.3 Les identifiants du dépôt logiciel

5.3.1 Référencer avec le SWHID (SoftWare Heritage Identifiers)

 archived `swh:1.dir:ec4ae097465d9ea51589537ea94b2ea50e8d134d`

- ★ identification de l'objet logiciel
- ★ une empreinte digitale d'une version spécifique

Nécessaire pour :

- **identifier - reproduire**
- **archiver**

5.3.2 Citer avec le HAL-ID

`hal-02309043, version 1`

- ★ identification d'une notice
- ★ métadonnées avec le dépôt
- ★ auteurs et contributeurs sont vérifiés en modération

Nécessaire pour :

- **créditer les auteurs**
- **indexer**

5.4 La citation et l'export BibTeX

- Citation accessible sur la notice
- Export BibTeX en utilisant le type d'entrée [BibLaTeX](#) **@software** ou **@softwareversion** (si un numéro de version a été renseigné)
- Export utilisé sur le rapport d'activité scientifique d'Inria depuis 2020.

Softwares

[38] [SW] M. Frigo, M. Zucchelli, R. Deriche and S. Deslauriers-Gauthier, *TALON: Tractograms As Linear Operators in Neuroimaging* version 0.3.0, 19th Jan. 2021. HAL: [\(hal-03116143\)](#), URL: <https://hal.archives-ouvertes.fr/hal-03116143>, VCS: <https://gitlab.inria.fr/cobcom/talon>, SWHID: [swh:1:dir:f25157ad1b13cb20ac3457d4f6756b49ac63d079;origin=https://hal.archives-ouvertes.fr/hal-03116143;visit=swh:1:snp:465d89956196578717f4cb515e456c279aa6a22;anchor=swh:1:rev:10247a14640a280b9140a27ce003d382d70cccac;path=/](https://hal.archives-ouvertes.fr/hal-03116143;visit=swh:1:snp:465d89956196578717f4cb515e456c279aa6a22;anchor=swh:1:rev:10247a14640a280b9140a27ce003d382d70cccac;path=/).

Le format citation sur HAL

Matteo Frigo, Mauro Zucchelli, Rachid Deriche, Samuel Deslauriers-Gauthier. TALON: Tractograms As Linear Operators in Neuroimaging. 2021.

[/swh:1:dir:f25157ad1b13cb20ac3457d4f6756b49ac63d079;origin=https://hal.archives-ouvertes.fr/hal-03116143;visit=swh:1:snp:465d89956196578717f4cb515e456c279aa6a22;anchor=swh:1:rev:10247a14640a280b9140a27ce003d382d70cccac;path=/](https://hal.archives-ouvertes.fr/hal-03116143;visit=swh:1:snp:465d89956196578717f4cb515e456c279aa6a22;anchor=swh:1:rev:10247a14640a280b9140a27ce003d382d70cccac;path=/).
([hal-03116143](#))

```
@softwareversion{frigo:hal-03116143v1,
  TITLE = {{TALON: Tractograms As Linear Operators in
Neuroimaging}},
  AUTHOR = {Frigo, Matteo and Zucchelli, Mauro and
Deriche, Rachid and Deslauriers-Gauthier, Samuel},
  URL = {https://hal.archives-ouvertes.fr/hal-03116143},
  NOTE = {},
  YEAR = {2021},
  MONTH = Jan,
  SWHID =
{swh:1:dir:f25157ad1b13cb20ac3457d4f6756b49ac63d079;origin
=https://hal.archives-ouvertes.fr/hal-03116143;visit=swh:1
:snp:465d89956196578717f4cb515e456c279aa6a22;anchor=swh:1
:rev:10247a14640a280b9140a27ce003d382d70cccac;path=/},
  VERSION = {0.3.0},
  REPOSITORY = {https://gitlab.inria.fr/cobcom/talon},
  LICENSE = {MIT License},
  KEYWORDS = {diffusion MRI ; dMRI ; tractography ; python
; optimization},
  FILE =
{https://hal.archives-ouvertes.fr/hal-03116143/file/talon-
source.zip},
  HAL_ID = {hal-03116143},
  HAL_VERSION = {v1},
}
```

Références

- ❖ Y. Barborini, R. Di Cosmo, Antoine R. Dumont, M. Gruenpeter, B. Marmol, A. Monteil, J. Sadowska.. La création du nouveau type de dépôt scientifique - Le logiciel. *JSO 2018 - 7es journées Science Ouverte Couperin : 100 % open access : initiatives pour une transition réussie*, Jan 2018, Paris, France. 2018. ([hal-01688726](#))
- ❖ R. Di Cosmo, M. Gruenpeter, B. Marmol, A. Monteil, L. Romary, J. Sadowska. *Curated Archiving of Research Software Artifacts: lessons learned from the French open archive*. IJDC. 2020 ([10.2218/ijdc.v15i1.698](#)). ([hal-02475835](#))
- ❖ R. Di Cosmo, M. Gruenpeter, S. Zacchiroli *Referencing Source Code Artifacts: a Separate Concern in Software Citation*, CiSE, IEEE, pp.1-9. 2020. ([10.1109/MCSE.2019.2963148](#)) ([hal-02446202](#))
- ❖ P. Alliez, R. Di Cosmo, B. Guedj, A. Girault, M.-S. Hacid, et al.. *Attributing and Referencing (Research) Software: Best Practices and Outlook from Inria*. Computing in Science and Engineering, Institute of Electrical and Electronics Engineers, 2019, pp.1-14. ([10.1109/MCSE.2019.2949413](#)). ([hal-02135891](#))
- ❖ A. Monteil, M. Gruenpeter, J. Sadowska, E. Nivault. *Garantir la cohérence des données constitue le cœur de notre activité: entretien autour des enjeux descriptifs du code source*. *Bulletin des bibliothèques de France*, Ecole Nationale Supérieure des Sciences de l'Information et des Bibliothèques (ENSSIB), 2021, Dossier BBF 2021-1 • Code source : libérer le patrimoine !. ([hal-03239502](#))