



FAQ du modérateur de logiciel dans HAL

	<i>Dans quels cas de figure contacter cet interlocuteur ?</i>	<i>Contact</i>
CCSD	Infos sur les fonctionnalités de HAL, les formations, etc.	hal.support@ccsd.cnrs.fr
SWH	Infos sur SWH	deposit@softwareheritage.org

Sommaire

Sigles et acronymes utilisés.....	3
Questions.....	3
I. [dépôt source] Dans le champ « Fichier(s) », je vois que le déposant a archivé plusieurs fichiers au lieu d'un seul dossier zippé. Puis-je commencer à vérifier les métadonnées ?.....	3
II. [dépôt source] Un déposant me demande quelles sont les extensions acceptées pour effectuer un dépôt source ?.....	3
III. [dépôt source] Un déposant peut-il fournir un fichier CodeMeta.json dans le cadre d'un dépôt source ?.....	3
IV. [dépôt source] Comment avoir la certitude que les fichiers déposés correspondent bien à du code source ?.....	3
V. [SWHID] Un déposant m'indique que son SWHID ne fonctionne pas quand il le copie-colle dans l'interface HAL. Comment l'aider ?	4
Ressource utile	5
VI. [SWHID] Pourquoi y a-t-il un risque d'erreur sur le type de SWHID fourni par le déposant en cas de mise à jour ? Pourquoi le modérateur doit-il vérifier ce champ quand le dépôt est de nouveau à modérer ?	5
Exemple de SWHID correct pour un dépôt dans HAL	6
Exemple de SWHID inadapté au dépôt dans HAL	6
Exemple de SWHID inadapté au dépôt dans HAL	7
Ressource utile	7
VII. [SWHID-HAL-ID] Quelles sont les différences entre un SWHID et un HAL-ID ?	8
VIII. [versions] Comment gérer plusieurs versions d'un logiciel dans HAL ? Quelles actions prévoir en tant que modérateur ?	9
IX. [versions] Un déposant signale un changement d'auteur pour la nouvelle version du logiciel qu'il souhaite déposer dans HAL. Faut-il procéder comme s'il s'agissait d'une nouvelle version du document ou comme s'il s'agissait du premier dépôt d'un nouveau document ?.....	10
X. [versions] [dépôt source] [dépôt SWHID] Peut-on recourir aux 2 méthodes de dépôt pour un même logiciel ?.....	10



- XI. [description] Le déposant n'a pas fourni de fichier CodeMeta.json, mais un fichier Zenodo.json, qui semble contenir le même type d'informations. Faut-il demander au déposant de prévoir aussi un CodeMeta.json ? 11
- XII. [description] Dans HAL, la plupart des champs sont facultatifs. Existe-t-il une norme descriptive en matière de logiciel, un vocabulaire contrôlé dédié ? Et en tant que modérateur, quelles bonnes pratiques adopter ? 11
- XIII. [HAL-SWH] Ai-je besoin de me créer un compte sur SWH pour modérer un dépôt logiciel ? 12
- XIV. [HAL-SWH] Y a-t-il des différences selon que le déposant archive son code source d'abord dans HAL ou d'abord dans SWH ou cela revient-il à la même chose compte tenu que les plateformes sont connectées ? 13
- XV. [HAL-SWH] Un même logiciel est archivé à la fois dans SWH et dans HAL mais la notice HAL ne comporte pas de SWHID. Que dois-je faire en tant que modérateur ? Et à quoi est-ce dû ? 14
- XVI. [HAL-SWH] Dans le cas d'un dépôt source mis en ligne sur HAL mais en attente de SWHID, que se passe-t-il pendant ce délai ? Le dépôt est-il visible sur HAL ? 14
- XVII. [HAL-SWH] D'un point de vue technique, comment SWH et HAL échangent-elles ? 15
- XVIII. [HAL-SWH] En fin de modération, quand on met en ligne le dépôt, le SWHID n'apparaît pas immédiatement. Pour différentes raisons, je veux faire le suivi de cette information, comment procéder ? 15
- XIX. [HAL-SWH] Dans l'interface publique HAL, qu'est-ce qui est décompté en « téléchargements de fichier » (vs « consultation de la notice »), compte tenu que même dans le cas d'un dépôt source, le dossier zippé est remplacé à terme par un SWHID ? 16
- XX. [HAL-SWH] Les dépôts SWHID n'ont pas de fichier associé : dans HAL, sont-ils considérés comme des notices ou comme des documents ? 16
- XXI. [SWH] Peut-on faire des développements dans SWH ? 16
- XXII. [SWH] Vers quoi renvoie le lien en rouge, en haut de page sur SWH ? 17
- XXIII. [doublon] Peut-il y avoir des doublons dans SWH ? 17
- XXIV. [doublon] Comment savoir si le logiciel a déjà été déposé dans HAL ? 17
- XXV. [publications] Quand le champ « Publications associées » est complété, la référence est-elle cliquable ? 17
- XXVI. [publications] Quand il y a une publication associée au logiciel, peut-on compléter la notice de la publication pour qu'elle renvoie vers celle du logiciel ? Le modérateur peut-il effectuer directement cette tâche ? 17
- XXVII. [publications] Dans CodeMeta, il y a le champ « referencePublication » : peut-on indiquer plusieurs références dans ce champ ? 18
- XXVIII. [diffusion] Un logiciel peut-il être sous embargo ? Un déposant me dit qu'il ne souhaite pas diffuser le code source avant un délai déterminé. Que faire en tant que modérateur ? 18
- XXIX. [auteurs] Quelle est la différence entre un auteur et les autres rôles disponibles dans HAL ? 19



XXX.	[politique de dépôt] Un chercheur me demande s'il serait possible de déposer le code source produit par un étudiant. Quelle est la politique en matière de dépôt dans HAL et dans SWH ?	19
XXXI.	[politique de dépôt] Un chercheur souhaite qu'un doctorant dépose pour lui son code source sur HAL. Est-ce possible ? Qui peut être déposant ?	19
XXXII.	Auteurs	19

Sigles et acronymes utilisés

- CCSD : Centre pour la Communication Scientifique Directe
- IES-Inria : Information et édition scientifique Inria
- MD : métadonnées
- SWH : Software Heritage
- SWHID : Software Heritage Identifier

Questions

- I. [dépôt source] Dans le champ « Fichier(s) », je vois que le déposant a archivé plusieurs fichiers au lieu d'un seul dossier zippé. Puis-je commencer à vérifier les métadonnées ?

Non : il faut contacter le déposant pour lui demander de rassembler tous ses fichiers dans une seule archive zippée. Il n'y a pas de travail dans HAL à prévoir tant que ce point n'est pas traité. Si le problème persiste, contactez le CCSD.

- II. [dépôt source] Un déposant me demande quelles sont les extensions acceptées pour effectuer un dépôt source ?

Variante : « Un déposant m'indique qu'il rencontre des problèmes de compatibilité d'extension pour déposer ses fichiers de code dans les fichiers annexes. »

La question des formats de fichiers ne se pose pas car le déposant doit fournir tous ses fichiers et dossiers dans une archive zippée. Dans le cadre d'un dépôt source, le déposant n'utilise pas les fichiers annexes pour fournir le code source. Et dans le cadre d'un dépôt SWHID, le déposant n'a pas de fichiers à déposer sur HAL : il précise uniquement l'identifiant SWHID et complète les MD manquantes.

- III. [dépôt source] Un déposant peut-il fournir un fichier CodeMeta.json dans le cadre d'un dépôt source ?

Oui, techniquement, c'est possible. L'avantage est que ces fichiers de métadonnées descriptives sont inclus avec les autres fichiers constitutifs du dépôt.







- IV. [dépôt source] Comment avoir la certitude que les fichiers déposés correspondent bien à du code source ?

Une extension de fichier correspondant à un format de vidéo indique qu'il y a une erreur de type de document pour le dépôt. Mais au-delà de ce cas, l'analyse de l'extension de fichier seule ne suffit pas forcément à trancher. Plusieurs questions peuvent se poser sur la nature de la ressource déposée dans HAL lors d'un dépôt source :

- Le déposant dépose sous la catégorie « Logiciel » une ressource qui n'est pas un logiciel : ex. base de données, jeux de données, data paper, etc.



- Inversement, le modérateur peut penser que le déposant a fait une erreur de catégorie de document alors qu'il s'agit bien d'un logiciel : ex. tableur avec des macros. Dans la plupart des cas, un fichier Excel ne contient que de la donnée. Mais il est possible d'avoir dans un tableur Excel une procédure appelée MACRO. En elle-même, cette procédure est un logiciel. Tout dépend de la description et du README du déposant.

Infos à vérifier dans le fichier README	Actions pour le modérateur
Il y a une mention type « software », « code », « procédure », « workflow » ou tout autre mot qui indique une action de transformation des données.	 Après une première analyse, contacter le déposant pour lui faire part de vos conclusions et obtenir une confirmation : ex. « D'après les informations incluses dans le README, je comprends qu'il s'agit bien d'un logiciel mais pour en avoir la certitude, j'aurais besoin d'une confirmation de votre part pour entamer la vérification des métadonnées ».
On n'évoque que les données, sans référence à un processus de transformation. Il peut s'agir de données brutes comme de données traitées.	 Ne pas entamer la vérification des métadonnées.  Contacter le déposant : Ex. « Sur la base des informations dont je dispose, il se pourrait que la ressource déposée ne soit pas de type « logiciel ». Il semblerait plutôt que ce sont des données. J'aurais besoin d'une confirmation de votre part pour faire le nécessaire. »
Aucun élément ne permet de trancher.	 Ne pas entamer la vérification des métadonnées.  Contacter le déposant pour collecter des informations complémentaires : Ex. « Afin de lui garantir une visibilité optimale et d'appliquer la procédure adaptée, je souhaiterais m'assurer auprès de vous que la ressource déposée est bien un logiciel. En effet, j'ai besoin de savoir s'il s'agit d'un outil permettant de transformer des données ou d'un recueil de données. »  En cas de doute, une fois les informations collectées auprès du déposant, contacter : hal.support@ccsd.cnrs.fr

V. [SWHID] Un déposant m'indique que son SWHID ne fonctionne pas quand il le copie-colle dans l'interface HAL. Comment l'aider ?

Cela peut venir de la manière dont le copier-coller est effectué depuis SWH.



Bonne pratique : utiliser le bouton « Copy identifier » plutôt que sélectionner le texte sur la page ou reprendre l'URL de la barre d'adresse du navigateur. Copier-coller le texte sans passer par le bouton peut créer des retours à la ligne.

To reference or cite the objects present in the Software Heritage archive, permalinks based on SoftWare Heritage persistent IDentifiers (SWHIDs) must be used instead of copying and pasting the url from the address bar of the browser (as there is no guarantee the current URI scheme will remain the same over time).

Select below a type of object currently browsed in order to display its associated SWHID and permalink.

directory revision snapshot

archived repository archived swh:1:dir:07577fdbb9f84684d86e345f50252f13134b2f30

iframe embedding

```
swh:1:dir:07577fdbb9f84684d86e345f50252f13134b2f30;
origin=https://github.com/tapaskumar0midwest/tapaskumar0midwest.github.io;
visit=swh:1:snp:1b09bc7f69089455d9dbbb457fd3701147228691;
anchor=swh:1:rev:f20b9980efecfe6f34b7b7a86c4c7ac223f4913b
```

Add contextual information

Utiliser le bouton

Copy identifier Copy permalink

Ne pas copier-coller le texte

Figure 1 : Interface de SWH

Le modérateur peut aussi s'assurer que le déposant a sélectionné le type d'identifiant SWHID dans HAL. Par défaut, quand l'utilisateur est dans la partie « Chargez les métadonnées à partir d'un identifiant », c'est le DOI qui est présélectionné.

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.

DOI 10.xxx

Récupérer les métadonnées

DOI
arXiv
PubMed
PubMed Central
SWHID
Bibcode ADS
CERN
PPN (Sudoc)
inSPIRE HEP
OATAO
Horizon (IRD)

Métadonnées du document

Auteurs

Ressource utile

Un GIF vaut mille mots : cette ressource montre comment obtenir un SWHID : https://annex.softwareheritage.org/public/tutorials/getswhid_dir.gif

VI. [SWHID] Pourquoi y a-t-il un risque d'erreur sur le type de SWHID fourni par le déposant en cas de mise à jour ? Pourquoi le modérateur doit-il vérifier ce champ quand le dépôt est de nouveau à modérer ?

Il existe plusieurs types de SWHID en fonction du type d'artefact numérique référencé dans SWH, c'est pourquoi le modérateur doit vérifier qu'il s'agit bien d'un SWHID avec le préfixe « dir » pour « directory ».

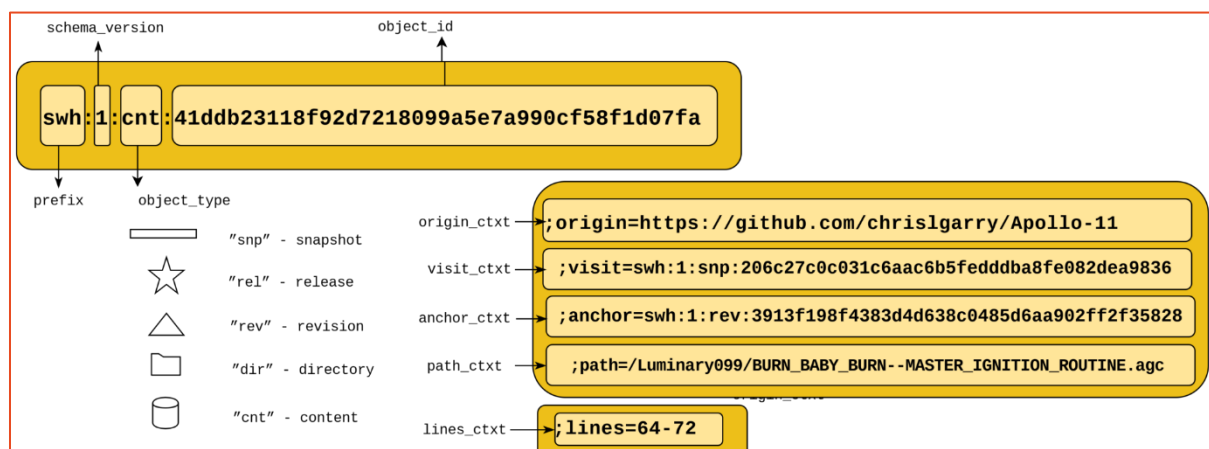


Figure 2 : détail de la syntaxe des SWHID. Dans cet exemple, il s'agit d'une vue détaillée d'un SWHID de type « Content », d'où le préfixe « cnt ». Dans le cadre du dépôt dans HAL, les préfixes « dir » sont requis.

Démarche générale :

- Vérifier que le préfixe est « dir » (et non pas « snp », « rel », « rev », « cnt »)
- Vérifier que les informations contextuelles sont ajoutées : dans le schéma ci-dessus de la syntaxe détaillée, il s'agit des éléments taggés « ctxt » « origin », « visit », « anchor », « path » (« lines » n'apparaît pas systématiquement).

Exemple de SWHID correct pour un dépôt dans HAL

swh:1:dir:ec88e5b901c034d5a91aa133e824d65cff3788a3;origin=https://github.com/rdicosmo/param;visit=swh:1:snp:25490d451af2414b2a08ece0df643dfdf2800084;anchor=swh:1:rev:db44dc9cf7a6af7b56d8ebda8c75be3375c89282

Dans ce cas, on retrouve bien un SWHID pour un objet de type « directory » comme l'indique le préfixe et on identifie les informations contextuelles, surlignées ici en jaune.

Exemple de SWHID inadapté au dépôt dans HAL

swh:1:dir:ec88e5b901c034d5a91aa133e824d65cff3788a3

Dans ce cas, le type de SWHID est bien « directory » mais il manque les informations contextuelles.



Permalinks

To reference or cite the objects present in the Software Heritage archive, permalinks based on Software Heritage persistent IDentifiers (SWHIDs) must be used instead of copying and pasting the url from the address bar of the browser (as there is no guarantee the current URI scheme will remain the same over time).

Select below a type of object currently browsed in order to display its associated SWHID and permalink.

directory
 revision
 snapshot

archived repository
 archived swh: 1:dir:ec88e5b901c034d5a91aa133e824d65cff3788a3
 Iframe embedding

```

swh:1:dir:ec88e5b901c034d5a91aa133e824d65cff3788a3;
origin=https://github.com/rdicosmo/parmap;
visit=swh:1:snp:25490d451af2414b2a08ece0df643dfdf2800084;
anchor=swh:1:rev:db44dc9cf7a6af7b56d8ebda8c75be3375c89282
  
```

Add contextual information

Figure 3 : Pour obtenir les informations contextuelles du SWHID, il faut laisser cochée la case "Add contextual information" puis cliquer sur "Copy identifier"

Exemple de SWHID inadapté au dépôt dans HAL

swh:1:cnt:8722d84d658e5e11519b807abb5c05bfbfc531f0;origin=https://github.com/rdicosmo/parmap;visit=swh:1:snp:25490d451af2414b2a08ece0df643dfdf2800084;anchor=swh:1:rev:db44dc9cf7a6af7b56d8ebda8c75be3375c89282;path=/example/mandels.ml

Dans ce cas, les informations contextuelles sont présentes, mais le type de SWHID est « content » (préfixe « cnt ») au lieu de « directory » (« dir »).

Il faut demander au déposant un autre SWHID.

Permalinks

To reference or cite the objects present in the Software Heritage archive, permalinks based on Software Heritage persistent IDentifiers (SWHIDs) must be used instead of copying and pasting the url from the address bar of the browser (as there is no guarantee the current URI scheme will remain the same over time).

Select below a type of object currently browsed in order to display its associated SWHID and permalink.

content
 directory
 revision
 snapshot

archived repository
 archived swh: 1:cnt:8722d84d658e5e11519b807abb5c05bfbfc531f0

Le déposant qui a fourni un SWHID de type « content » peut obtenir un SWHID type « directory » en utilisant cet onglet, depuis SWH.

Ressource utile

Un GIF vaut mille mots : cette ressource montre comment obtenir un SWHID : https://annex.softwareheritage.org/public/tutorials/getswhid_dir.gif



VII. [SWHID-HAL-ID] Quelles sont les différences entre un SWHID et un HAL-ID ?

Pour comprendre la différence entre SWHID et HAL-ID, il faut faire la différence entre référencement et citation de logiciel.

	Référencer	Citer
<i>Objectif</i>	Prendre l'empreinte numérique des artefacts qui résultent d'un développement logiciel (ex. code source, binaires exécutables, etc.), afin d'en assurer la réutilisation. Le travail d'identification doit porter sur les différents niveaux de granularité.	Identifier les auteurs d'un projet logiciel, attribuer le crédit d'une création à des auteurs.
	On décrit des objets numériques.	On décrit le projet qui a permis d'aboutir à des objets numériques, mais le projet en lui-même n'est pas un objet numérique. On attribue le crédit d'une œuvre à des auteurs : c'est pour cette raison qu'une étape de vérification manuelle est nécessaire.
	Les 2 objectifs sont liés mais pas dépendants l'un de l'autre. Par exemple, pour assurer la reproductibilité d'une recherche, référencer les artefacts du logiciel suffit.	
<i>Identifiant mobilisé dans le cadre du dépôt</i>	SWHID	HAL-ID
<i>Nécessaire pour</i>	Identifier, reproduire la recherche (on sait quelle version précise d'un artefact a été utilisée par l'auteur d'un article) Archiver	Créditer les auteurs Indexer
Source : Gruenpeter, Morane, Jozefina Sadowska, Estelle Nivault, and Alain Monteil. 'Create Software Deposit in HAL'. Technical Report. Inria ; CCSD ; Software Heritage, 2022. https://hal.inria.fr/hal-01872189 .		

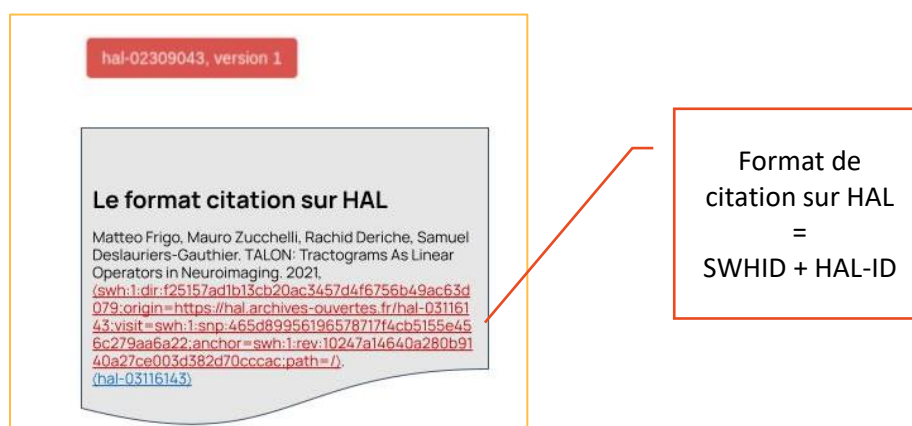


Figure 4 : Di Cosmo, R., Gruenpeter, M., Monteil, A., & Marmol, B. (2022, May 31). Les logiciels dans HAL. Une collaboration entre Software Heritage, Inria et le CCSD. Parlons science ouverte, Online Conference.

<https://docs.google.com/presentation/d/1n7NyPXsnQr>

Le SWHID est un identifiant intrinsèque permanent permettant l'identification sur le long terme des artefacts d'un logiciel, qui sont des objets numériques produits lors du processus de développement. Le SWHID a été conçu pour répondre aux besoins spécifiques de l'identification des logiciels.

A la différence des identifiants extrinsèques comme les DOI, les ISSN ou les ISBN, créer un SWHID ne nécessite pas le recours à une agence (ex. DataCite, CrossRef), ni même à SWH¹. L'identifiant est créé automatiquement selon des techniques issues de la cryptographie qui assurent l'intégrité des identifiants : en d'autres termes, un identifiant reste associé à un objet de manière durable et transparente ; un changement ne peut pas intervenir aux dépens de l'utilisateur. Enfin, attribuer des SWHID ne nécessite pas d'abonnement auprès d'une agence.

VIII. [versions] Comment gérer plusieurs versions d'un logiciel dans HAL ? Quelles actions prévoir en tant que modérateur ?

Voici l'exemple d'un logiciel avec plusieurs versions : <https://hal.inria.fr/hal-03643569v2>

La procédure de dépôt d'une nouvelle version dans HAL ne change pas en elle-même : il s'agit soit d'un dépôt source, soit d'un dépôt SWHID. Et sur l'interface publique de HAL, comme pour les autres types de documents, l'ensemble des versions est accessible.

Attention, le rythme de mise à jour dans HAL ne correspond pas forcément au rythme de mise à jour sur une plateforme de développement. Par exemple, la version 2 du logiciel dans HAL peut correspondre à la version 4 du logiciel en termes techniques. **La version dans HAL est une version de notice et pas une version de logiciel.**

Si un déposant s'interroge sur la fréquence de la mise à jour de dépôt à prévoir dans HAL, le modérateur peut lui conseiller de prévoir un dépôt d'une nouvelle version en cas de changement majeur, pas à chaque modification du contenu du code source.

¹ « [...] SWH-IDs are intrinsic identifiers is that they can be generated and verified independently of Software Heritage, using swh-identify, an open source tool developed by Software Heritage, and distributed via PyPI as swh.model, with the stable version at the time of writing being this one. » Di Cosmo, Roberto. 'Archiving and Referencing Source Code with Software Heritage'. In *Mathematical Software – ICMS 2020*, edited by Anna Maria Bigatti, Jacques Carette, James H. Davenport, Michael Joswig, and Timo de Wolff, 362–73. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2020. https://doi.org/10.1007/978-3-030-52200-1_36.



Le SWHID est associé à une version spécifique des contenus : son rôle n'est pas de pointer vers une version changeante du code source.

IX. [versions] Un déposant signale un changement d'auteur pour la nouvelle version du logiciel qu'il souhaite déposer dans HAL. Faut-il procéder comme s'il s'agissait d'une nouvelle version du document ou comme s'il s'agissait du premier dépôt d'un nouveau document ?

La priorité est de contacter le déposant pour s'assurer qu'il s'agit bien d'une évolution du logiciel qui était précédemment décrit dans la notice. Si le déposant confirme, on considère qu'il s'agit d'une nouvelle version d'un même document et non pas d'un nouveau dépôt.

D'une version à l'autre, les contributeurs et les auteurs d'un logiciel peuvent changer.


En revanche, ce cas est à distinguer du cas d'un changement significatif des métadonnées ; ex. le titre a changé. Avant de contacter le déposant pour obtenir des informations complémentaires, le modérateur peut comparer les différents README pour voir si par exemple, les fonctionnalités et finalités du logiciel sont différentes d'une version à l'autre.

X. [versions] [dépôt source] [dépôt SWHID] Peut-on recourir aux 2 méthodes de dépôt pour un même logiciel ?

Oui, dans le cas du dépôt d'une nouvelle version du logiciel. Par exemple, le déposant effectue un dépôt source pour la V1 puis un dépôt SWHID pour la V2. Compte tenu que les contenus du code repository ont changé, le déposant fournit un nouveau SWHID pour la V2.

Non, dans le cas des étapes de dépôt d'une même version. C'est-à-dire qu'au cours d'un même dépôt, on ne peut pas passer d'une méthode à l'autre ou fournir d'emblée un SWHID et un dossier zippé : la nouvelle version de HAL rend impossible ce type de dépôt.

Toutefois, si un même dépôt est composé d'un SWHID et d'un dossier zippé, voici les étapes à prévoir par le modérateur :

 Contacter le déposant pour lui demander s'il y a une différence de contenu entre ce vers quoi renvoie le SWHID et le contenu du dossier zippé.	
Réponse du déposant : « Oui, il y a une différence de contenu. »	Réponse du déposant : « Non, il n'y a pas de différence de contenu. »
Le modérateur demande au déposant de fournir des précisions sur l'objectif qu'il cherche à atteindre afin de déterminer s'il ne faut pas plutôt procéder à 2 dépôts distincts.	Le modérateur invite le déposant à choisir ce qu'il souhaite conserver en lui donnant des éléments d'aide à la décision : <ul style="list-style-type: none"> - conserver le SWHID et pas le dossier zippé = conserver l'historique de développement du code source - conserver le dossier zippé et pas le SWHID = conserver le code source sans son historique de développement La décision appartient au déposant.



- XI. [description] Le déposant n'a pas fourni de fichier CodeMeta.json, mais un fichier Zenodo.json, qui semble contenir le même type d'informations. Faut-il demander au déposant de prévoir aussi un CodeMeta.json ?

Non. Le modérateur peut s'appuyer sur les informations du fichier Zenodo.json ou de tout autre type de fichier (ex. fichier CFF – Citation file format). Ce qui compte dans la modération HAL est de disposer de sources d'information permettant de vérifier les métadonnées dans HAL. Si le modérateur trouve l'information qui lui est nécessaire dans les fichiers fournis, il n'a pas besoin de demander un dépôt de fichier complémentaire.

- XII. [description] Dans HAL, la plupart des champs sont facultatifs. Existe-t-il une norme descriptive en matière de logiciel, un vocabulaire contrôlé dédié ? Et en tant que modérateur, quelles bonnes pratiques adopter ?

Non, actuellement, il n'existe pas de norme descriptive et certaines informations ne sont pas vérifiables directement par le modérateur car elles requièrent une expertise avancée sur le logiciel concerné. La difficulté réside moins dans la surabondance de schémas de métadonnées que de l'absence de procédure unifiée. La description des logiciels est un terrain nettement moins balisé que celui des ouvrages ou des articles.

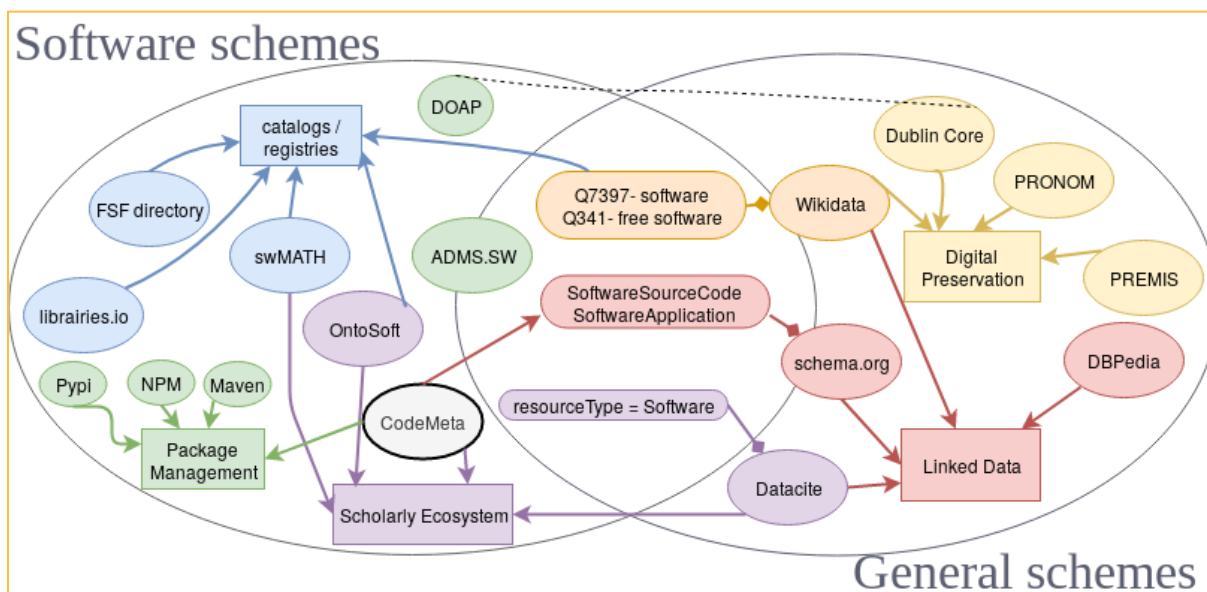


Figure 5 : Ce schéma illustre la multiplicité de schémas descriptifs pour les logiciels. CodeMeta présente l'avantage d'être une passerelle entre plusieurs schémas.

Source : Software Heritage. (n.d.). Data model. Software Heritage Development Documentation. Retrieved 23 August 2022, from <https://docs.softwareheritage.org/devel/swh-model/data-model.html>

Le modérateur intervient pour garantir la cohérence des données saisies dans HAL avec les sources fournies par le déposant.

Dans une perspective de sensibilisation plus générale, dans son établissement, le modérateur peut inciter les déposants potentiels à prévoir en amont, dès leur travail dans une plateforme de



développement collaboratif (ex. GitLab, GitHub, etc.), un ensemble de fichiers et d'informations qui faciliteront ensuite leur dépôt dans HAL :

Type de fichier	Exemple de ressource pour produire un fichier de qualité
CodeMeta.json	https://codemeta.github.io/codemeta-generator/
CFF	https://citation-file-format.github.io/cff-initializer-javascript/#/ Tutoriel pour utiliser cffinit, une application qui permet de créer un fichier CFF tout en étant guidé pour compléter les champs : https://the-turing-way.netlify.app/communication/citable/citable-cffinit.html#cm-citable-cffinit
README	https://www.makeareadme.com/ Possibilité pour le modérateur de prévoir une liste d'exemples de README bien rédigés

Rappel des MD HAL spécifiques au logiciel :

- SWHID
- Langage de programmation
- Code repository : lien où se trouve le développement du code (SVN, github, gitlab, CodePlex)
- Platform/OS : le système d'exploitation compatible avec le logiciel
- Version : version du logiciel déposé (peut être différente de la version publiée sur HAL)
- Etat du développement (development status) : l'état du développement du logiciel au moment du dépôt (Concept, WIP, Suspendu, Actif, Inactif)
- Outils de développement (runtime platform) : les outils de développement associés au logiciel (Framework, middleware, plateforme logicielle)

XIII. [HAL-SWH] Ai-je besoin de me créer un compte sur SWH pour modérer un dépôt logiciel ?

Non. Quel que soit le type de dépôt dans HAL, le modérateur n'a pas besoin de logins SWH :

- Dépôt source : le transfert vers SWH s'effectue depuis HAL et le SWHID est intégré automatiquement
- Dépôt SWHID : c'est le déposant qui travaille dans l'interface SWH. Les informations dont le modérateur a besoin pour vérifier les MD sont accessibles sans authentification.



XIV. [HAL-SWH] Y a-t-il des différences selon que le déposant archive son code source d'abord dans HAL ou d'abord dans SWH ou cela revient-il à la même chose compte tenu que les plateformes sont connectées ?

Oui, il existe des différences.

	Qu'est-ce qui est archivé ?
Dépôt source dans HAL puis export vers SWH	Le code source, sans son historique de développement = code source tel qu'il est au moment où le déposant crée le dossier zippé contenant les fichiers.
Dépôt SWHID dans HAL	Le code source avec son historique de développement = code source avec l'historique de développement tel qu'il est au moment où le déposant l'archive dans SWH. Cette méthode de dépôt est à conseiller aux utilisateurs d'outils de contrôle de version : Gitlab, GitHub, etc.

On peut conseiller aux utilisateurs les bonnes pratiques suivantes :

- Utilisation d'une plateforme de développement collaboratif pour bénéficier d'un outil de contrôle de version ; ex. GitLab, GitHub, Mercurial, etc.
- Si l'entrepôt (« repository ») créé sur la plateforme de développement est public, vérifier s'il est déjà sur SWH. (Mode d'emploi pour consulter SWH : « Home page » > « Archive » > « Browse » > « [Help](#)² »)
 - Si oui : archiver sur HAL le code source par un dépôt SWHID
 - Si non :
 - archiver le code source sur SWH en utilisant la fonction « Save Code Now » (disponible sous forme d'[extension pour Firefox et Chrome](#)³ et [sur l'interface de SWH](#)⁴)
 - archiver sur HAL le code source par un dépôt SWHID

² <https://archive.softwareheritage.org/>

³ <https://www.softwareheritage.org/2022/08/02/updateswh-browser-extension/>

⁴ <https://archive.softwareheritage.org/save/>



- XV. [HAL-SWH] Un même logiciel est archivé à la fois dans SWH et dans HAL mais la notice HAL ne comporte pas de SWHID. Que dois-je faire en tant que modérateur ? Et à quoi est-ce dû ?

Le modérateur n'a rien de particulier à prévoir dans ce cas de figure. Cela peut être dû au fait que le déposant a décoché la case du transfert vers SWH lors du dépôt dans HAL, mais que par ailleurs, le code source a été publié dans SWH.

Vue de la notice HAL du logiciel HBP Model Validation Framework

hal-03330341, version 1

MÉTADONNÉES

Shalesh Appukkuttan¹, Andrew P. Davison¹ Détails

¹ NeuroPSI - Institut des Neurosciences Paris-Saclay

Abstract: The Human Brain Project Model Validation Framework provides a REST API and a web app for searching, browsing and viewing the results of model validation experiments. This is version 1.7.

Type de document : **Logiciel**

Domaine : Informatique [cs] / Modélisation et simulation Sciences du Vivant [q-bio] / Neurosciences [q-bio.NC]

Liste complète des métadonnées [Voir](#)

TÉLÉCHARGER

Fichier : hbp-validation-framework-1.7.zip [Télécharger](#)

1.7

Licenses : [Apache License 2.0](#)

Langage de programmation : [Python](#) | [Java](#) | [React](#)

Code de dépôt : <https://github.com/HumanBrainProject/hbp-validation-framework>

COLLECTIONS

CNRS | TDS-MACS | UNIV-PARIS-SACLAY | GS-LIFE-SCIENCES-HEALTH | GS-HEALTH-DRUG-SCIENCES

CITATION

Shalesh Appukkuttan, Andrew P. Davison, HBP Model Validation Framework v1.7. 2021. (hal-03330341)

EXPORTER

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

[Endnote](#) [Datacite](#)

Alors que le dépôt est publié, il n'y a pas eu de transfert vers SWH depuis HAL comme l'indique le lien vers le dossier zippé.

Vue de la page SWH du logiciel HBP Model Validation Framework

<https://github.com/HumanBrainProject/hbp-validation-framework>

27 July 2022, 08:29:09 UTC

<> Code [Branches \(15\)](#) [Releases \(5\)](#) [Visits](#)

Revision: [eefa06591c53ee97c72758d877ec4b2725503b7](#) / [bc43f6b](#) / [History](#) [Download](#)

Tip revision: [eefa06591c53ee97c72758d877ec4b2725503b7](#) authored by Andrew Davison on 22 July 2022, 14:18:06 UTC

avoid problem with Harbor permissions

File	Mode	Size
------	------	------

Le logiciel est toutefois archivé dans SWH.

- XVI. [HAL-SWH] Dans le cas d'un dépôt source mis en ligne sur HAL mais en attente de SWHID, que se passe-t-il pendant ce délai ? Le dépôt est-il visible sur HAL ? Oui, sur HAL, la notice est consultable, avant même la mise en ligne sur SWH.



hal-01760120, version 1

Pareto Graph Contraction Algorithm

Philippe Nghe ¹, Bela M. Mulder ² [Details](#)

1 LBC, CBI UMR8231 - Laboratoire de Biochimie, Chimie-Biologie-Innovation, UMR8231 CNRS ESPCI ParisTech
2 FOM-Institute AMOLF

Python implementation of the Graph Contraction Algorithm described in Nghe P, Mulder B & Tans SJ, "A graph-based algorithm for the multi-objective optimization of gene regulatory networks" European Journal of Operational Research (2018)

Type de document : **Logiciel**

Domaine : **Sciences du Vivant [q-bio] / Bio-Informatique, Biologie Systémique [q-bio.QM]**

Liste complète des métadonnées

Format du dépôt	Fichier
Type de document	Logiciel
Titre	Pareto Graph Contraction Algorithm
Auteur(s)	Philippe Nghe ¹ , Bela M. Mulder ² 1 LBC, CBI UMR8231 - Laboratoire de Biochimie, Chimie-Biologie-Innovation, UMR8231 CNRS ESPCI ParisTech 252892 - 10 rue Vauquelin 75231 Paris Cedex 05 - France ESPCI ParisTech 301585 2 FOM-Institute AMOLF 198815 - Pays-Bas FOM-Institute AMOLF 348377
Résumé	Python implementation of the Graph Contraction Algorithm described in Nghe P, Mulder B & Tans SJ, "A graph-based algorithm for the multi-objective optimization of gene regulatory networks" European Journal of Operational Research (2018)
Version	version 1.0
Domaine	Sciences du Vivant [q-bio]Bio-Informatique, Biologie Systémique [q-bio.QM]
Licences	Apache License Version 2.0
Langage de programmation	python
Mots-clés	Pareto Algorithm, Gene regulatory Network, Python, graph-based algorithm

TÉLÉCHARGER

Fichier : ParetoGraphContraction-v1.0.zip

Sur l'interface publique de HAL, à ce stade, le dossier zippé est visible. Il n'y a pas encore de lien vers SWH.

XVII. [HAL-SWH] D'un point de vue technique, comment SWH et HAL échangent-elles ?

Le protocole Sword est utilisé. En savoir plus : [documentation technique](#)⁵

XVIII. [HAL-SWH] En fin de modération, quand on met en ligne le dépôt, le SWHID n'apparaît pas immédiatement. Pour différentes raisons, je veux faire le suivi de cette information, comment procéder ?

Dans le cadre d'un dépôt source, l'attribution d'un SWHID est automatique, mais ne déclenche pas une notification pour le modérateur.

Prévoir un tableur de suivi peut faciliter le travail : on note l'identifiant du dépôt, éventuellement le nom du logiciel et on vérifie 2 à 3 jours plus tard la notice.

⁵ <https://docs.softwareheritage.org/devel/swh-deposit/api/api-documentation.html>



- XIX. [HAL-SWH] Dans l'interface publique HAL, qu'est-ce qui est décompté en « téléchargements de fichier » (vs « consultation de la notice »), compte tenu que même dans le cas d'un dépôt source, le dossier zippé est remplacé à terme par un SWHID ?

hal-01897934, version 3

en
fr

SLALOM - Logiciel d'optimisation multivariée de cellules solaires

Sidi Ould Saad Hamady¹, Nicolas Fressengeas¹ [Détails](#)

1 LMOPS - Laboratoire Matériaux Optiques, Photonique et Systèmes

Abstract : SLALOM (SoLAr CeLL Multivariate OptiMizer) is a set of open-source Python programs implementing a rigorous mathematical methods for the optimization of solar cells using as backend a drift-diffusion device simulator. It aims to be simple to use, to maintain and to extend. It includes a core optimizer using the well tested robust mathematical methods, A set of user interface utilities and some complete and working examples easily adaptable to new solar cell technologies. SLALOM uses, as device simulator, the Silvaco(C) Atlas tool. It can be easily extended to use any simulator that have a standard input format and a command line interface.

Type de document : **Logiciel**

Domaine : **Physique [physics] / Matière Condensée [cond-mat] / Science des matériaux [cond-mat.mtrl-sci]**
Sciences de l'ingénieur [physics] / Electronique

Liste complète des métadonnées [Voir](#)

CONSULTER

Software Heritage [swh:1:dir:35c5f1a11f22bdab179e37e69bdca43b4640d6a1;origin=https://hal.archives-ouvertes.fr/hal-01897934;visit=swh:1:snp:441755d289c857cc32ea7dcbf1280139e3856379;anchor=swh:1:rev:f370e484d8579b4ccc863d1c2a2f10decad21914;path=/](https://hal.archives-ouvertes.fr/hal-01897934;visit=swh:1:snp:441755d289c857cc32ea7dcbf1280139e3856379;anchor=swh:1:rev:f370e484d8579b4ccc863d1c2a2f10decad21914;path=/)

[Consulter](#)

<https://hal.inria.fr/hal-01897934>
Contributeur : Sidi OULD SAAD HAMADY [Connectez-vous pour contacter le contributeur](#)
Soumis le : mardi 11 juin 2019 - 21:38:12
Dernière modification le : dimanche 5 juin 2022 - 19:06:06



Le nombre de téléchargements correspond au nombre de consultations de la page SWH du logiciel.

- XX. [HAL-SWH] Les dépôts SWHID n'ont pas de fichier associé : dans HAL, sont-ils considérés comme des notices ou comme des documents ?

Tous les dépôts logiciels issus d'un dépôt SWHID sont considérés comme des documents dans HAL.

- XXI. [SWH] Peut-on faire des développements dans SWH ?

Non. SWH n'est pas une plateforme de développement collaboratif. L'archive préserve le code source et son historique de développement à des fins de réutilisation, de traçabilité mais n'est pas un environnement technique pour des développeurs.

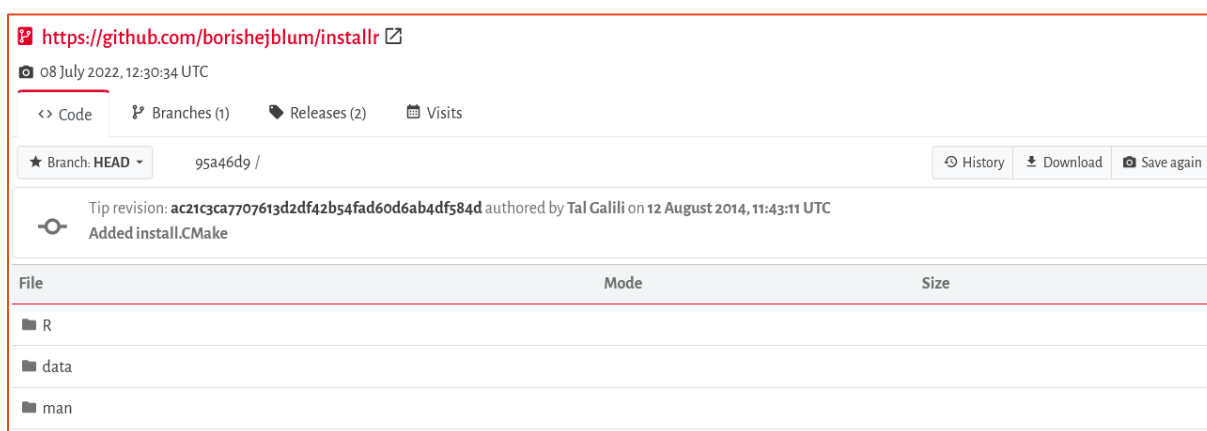


XXII. [SWH] Vers quoi renvoie le lien en rouge, en haut de page sur SWH ?

Sur l'interface de SWH, en haut de page, figure une URL qui renvoie vers la plateforme où le code source a été développé ; ici : <https://github.com/borishejblum/installr>.

En cliquant sur ce lien, on est ramené vers la version la plus récente de l'entrepôt de code (code repository) et non pas forcément vers la version du logiciel telle qu'elle a été déposée dans HAL. C'est pourquoi le lien vers la plateforme de développement initiale ne suffit pas et qu'il faut utiliser le SWHID qui seul permet d'identifier sans ambiguïté une version précise de l'entrepôt.

Les informations indiquées sous le lien en rouge précisent la date de sauvegarde la plus récente dans SWH et non pas la date des développements les plus récents.



XXIII. [doublon] Peut-il y avoir des doublons dans SWH ?

Non, un mécanisme de dédoublonnage tenant compte des différents niveaux de granularité des artefacts des logiciels est opérationnel : ainsi, un fichier identique contenu dans plusieurs projets n'est stocké qu'une fois.

« In particular, deduplication is built-in. Any software artifacts encountered in the wild gets added to the archive only if a corresponding node with a matching intrinsic identifier is not already available in the graph—file content, commits, entire directories or project snapshots are all deduplicated incurring storage costs only once » in Cosmo, Roberto Di, and Stefano Zacchiroli. 'Software Heritage: Why and How to Preserve Software Source Code', 1–10, 2017. <https://hal.archives-ouvertes.fr/hal-01590958>.

XXIV. [doublon] Comment savoir si le logiciel a déjà été déposé dans HAL ?

Avant de valider le dépôt, un message d'alerte s'affiche si des correspondances ont été détectées : noms d'auteurs, nom du logiciel, identifiant, etc.

XXV. [publications] Quand le champ « Publications associées » est complété, la référence est-elle cliquable ?

Oui, depuis fin septembre 2022, toutes les URL sont cliquables.

XXVI. [publications] Quand il y a une publication associée au logiciel, peut-on compléter la notice de la publication pour qu'elle renvoie vers celle du logiciel ? Le modérateur peut-il effectuer directement cette tâche ?

Oui, plusieurs documents peuvent être associés grâce à la fonctionnalité « Lier mes ressources » : <https://doc.archives-ouvertes.fr/tutoriels/liar-plusieurs-depots/>



Actuellement, une notice de logiciel peut renvoyer vers d'autres ressources mais la métadonnée « logiciel associé » n'existe pas dans les notices des autres types de documents.

Si techniquement, le modérateur peut associer des ressources, sur le plan de la méthode, il revient à chaque service de définir sa politique d'enrichissement des métadonnées. Attention, le modérateur n'a pas forcément les droits de modification sur le dépôt décrivant la publication.

XXVII. [publications] Dans CodeMeta, il y a le champ « *referencePublication* » : peut-on indiquer plusieurs références dans ce champ ?

Oui, on peut indiquer plusieurs références dans ce champ.

Et si le déposant utilise l'interface HAL (champ « Publications associées ») pour compléter cette information, le champ est répétable.

XXVIII. [diffusion] Un logiciel peut-il être sous embargo ? Un déposant me dit qu'il ne souhaite pas diffuser le code source avant un délai déterminé. Que faire en tant que modérateur ?

Sur le plan technique, oui, un logiciel peut être déposé dans HAL avec un embargo. Cette fonctionnalité est actuellement disponible pour un dépôt source et pas pour le dépôt SWHID. Il ne s'agit donc pas de faire créer une notice sans document associé. Durant la période d'embargo, la notice et le contenu associé ne sont pas visibles et ne sont pas transférés vers SWH.

Sur le plan de la méthode, si le logiciel n'a pas vocation à être exposé avant un certain délai, il est préférable de le déposer dans HAL une fois ce délai révolu. En effet, actuellement, la levée d'embargo dans HAL ne permet pas un transfert automatique vers SWH. Le code source ne serait donc ni archivé sur le long terme, ni associé à un identifiant pérenne SWHID.

Option : le modérateur peut inviter le déposant à rédiger un « *software paper* » permettant de décrire le logiciel, sans révéler son code source.

Romano, J. D., & Moore, J. H. (2020). Ten simple rules for writing a paper about scientific software. *PLOS Computational Biology*, 16(11), e1008390. <https://doi.org/10.1371/journal.pcbi.1008390>



XXIX. [auteurs] Quelle est la différence entre un auteur et les autres rôles disponibles dans HAL ?

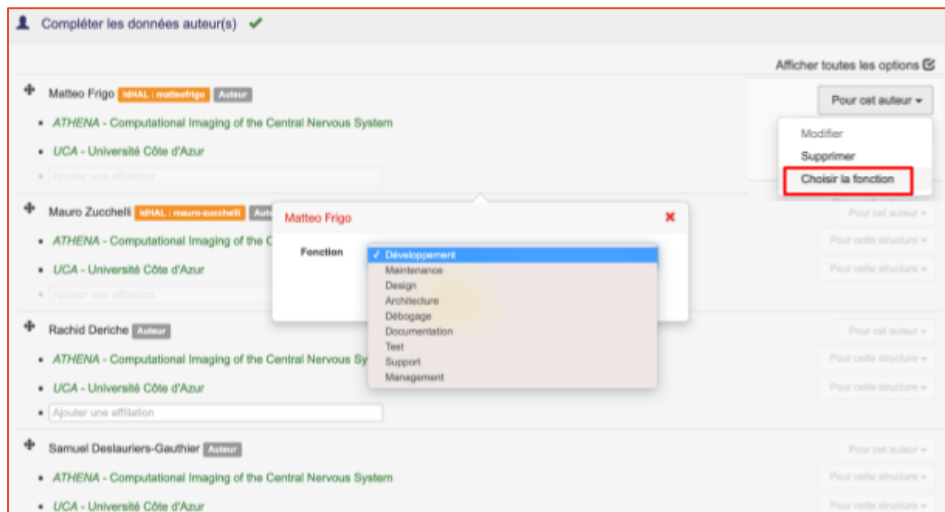


Figure 6 : Zoom sur les MD auteurs dans HAL : un rôle spécifique peut être attribué à chaque contributeur

Actuellement, ce sont essentiellement les auteurs qui sont indiqués dans les MD. Toutefois, HAL donne la possibilité aux déposants d'indiquer d'autres types de contributions : « maintenance », « design », « architecture », « débogage », « documentation », « test », « support », « management ».

Tous ces acteurs n'interviennent pas directement sur le code source mais jouent un rôle clé dans le développement d'un logiciel. Par exemple, les personnes impliquées dans l'architecture d'un logiciel travaillent sur les questions de maintenance, de modularité, d'efficacité. Il s'agit d'intégrer dès la phase de conception les enjeux de l'évolution sur le moyen et plus long terme.

Un déposant peut fournir la liste de tous les contributeurs dans un fichier « CONTRIBUTORS ».

XXX. [politique de dépôt] Un chercheur me demande s'il serait possible de déposer le code source produit par un étudiant. Quelle est la politique en matière de dépôt dans HAL et dans SWH ?

Oui, tous les codes sources sont acceptés. SWH n'effectue pas de sélection des codes sources.

XXXI. [politique de dépôt] Un chercheur souhaite qu'un doctorant dépose pour lui son code source sur HAL. Est-ce possible ? Qui peut être déposant ?

Seuls les créateurs/auteurs du logiciel ou leurs représentants peuvent déposer un logiciel sur HAL. Dans cet exemple précis, tout dépend du rôle que la personne mandatée a joué lors de la conception du logiciel : si le doctorant a contribué au code source, oui, il peut le déposer.

XXXII. Auteurs

Sabrina Granger : sabrina.granger@softwareheritage.org

Morane Gruenpeter : morane@softwareheritage.org

Alain Monteil : alain.monteil@inria.fr

Estelle Nivault : estelle.nivault@inria.fr

Jozefina Sadowska : josephine.sadowska@inria.fr