Ouvrir la science !

Vers un catalogue des logiciels issus de la recherche

État des lieux et analyse des besoins

Collège Codes sources et logiciels
Comité pour la science ouverte

Juillet 2024

Vers un catalogue des logiciels issus de la recherche

État des lieux et analyse des besoins

———

GT1 : Identification et mise en avant des logiciels de l'ESR

———

Violaine LOUVET (co-pilote GT1) CNRS / Université Grenoble Alpes

Sabrina GRANGER (co-pilote GT1) Inria / Software Heritage

Roberto DI COSMO (co-pilote du collège) Inria / Université Paris Cité

François PELLEGRINI (co-pilote du collège) Université de Bordeaux / Inria

———

Juillet 2024

———

DOI : 10.52949/79

Conception graphique : opixido

Résumé

L'un des objectifs majeurs du collège « Codes sources et logiciels » du comité pour la science ouverte est de contribuer à une meilleure compréhension du rôle du logiciel dans la recherche.

Le logiciel de recherche constitue une production scientifique de même importance que les publications et les données : ces trois facettes du savoir académique interagissent les unes avec les autres. De très nombreuses disciplines sont concernées, bien au-delà des sciences de l'informatique. Par « logiciel de recherche », on entend un logiciel créé, utilisé ou diffusé dans les productions de recherche. L'analyse reprend la définition du logiciel de recherche conçue par le collège.

Les communautés scientifiques, les laboratoires et les institutions ont besoin d'identifier et de rendre visibles leurs logiciels pour :

- garantir leur archivage et en pérenniser l'accès ;
- faciliter leur citation ;
- les valoriser, notamment sur le plan scientifique, afin de permettre leur réutilisation ;
- faire émerger de nouvelles collaborations.

Les logiciels développés dans les laboratoires de recherche étant de natures très diverses, il est nécessaire de mener une réflexion sur les critères de référencement et de conservation. Ainsi, les logiciels liés d'une manière ou d'une autre à une publication doivent être considérés comme particulièrement importants, notamment vis-à-vis des aspects de transparence et de reproductibilité. Les critères techniques ne permettent pas à eux seuls d'établir des priorités pertinentes : par exemple, un groupe restreint d'auteurs ou de contributeurs peut être à l'origine d'un logiciel peu complexe et de taille modeste mais dont l'impact académique est majeur. En d'autres termes, le degré de sophistication technique d'un logiciel, ou son nombre d'auteurs ou de contributeurs ne permettent pas, à eux seuls, d'augurer de son impact scientifique.

Les institutions et les auteurs de logiciels mobilisent déjà des canaux pour rendre leurs productions visibles : pages dédiées de sites web, plateformes de distribution, etc. Un catalogue de logiciels apporterait aux établissements comme aux individus des services complémentaires, notamment en fournissant un point d'entrée centralisé et des fonctionnalités plus riches que celles des moteurs de recherche généralistes. L'enjeu est double. D'une part, il s'agit d'inscrire la démarche dans un contexte élargi, afin d'identifier les infrastructures et initiatives existantes, en France comme à l'étranger. D'autre part, l'utilisateur final doit être placé au centre du processus : la charge de travail pour les chercheurs doit être réduite au maximum grâce à des processus d'automatisation.

Au-delà des enjeux techniques et fonctionnels, la problématique de l'accompagnement doit être prise en compte dès le début de la réflexion. De nombreux acteurs, aux profils et compétences variés, sont susceptibles de jouer un rôle en lien avec les logiciels. Outre les chercheurs et enseignants-chercheurs, on peut citer les personnels de soutien à la recherche, les personnels des bibliothèques et les équipes des services de valorisation. La dynamique actuelle autour des données de recherche offre un cadre sur lequel s'appuyer pour mettre en œuvre des actions d'accompagnement et renforcer une culture partagée du logiciel de recherche.

Enfin, une approche unifiée, tant sur le plan technique que sur celui de la gouvernance, permettra d'intégrer à la réflexion la question de la stratégie à adopter sur le long terme. Il s'agit en effet de positionner le projet de catalogue en cohérence avec l'écosystème de recherche français, à l'échelle des laboratoires comme des institutions.

Sommaire

1 | Introduction

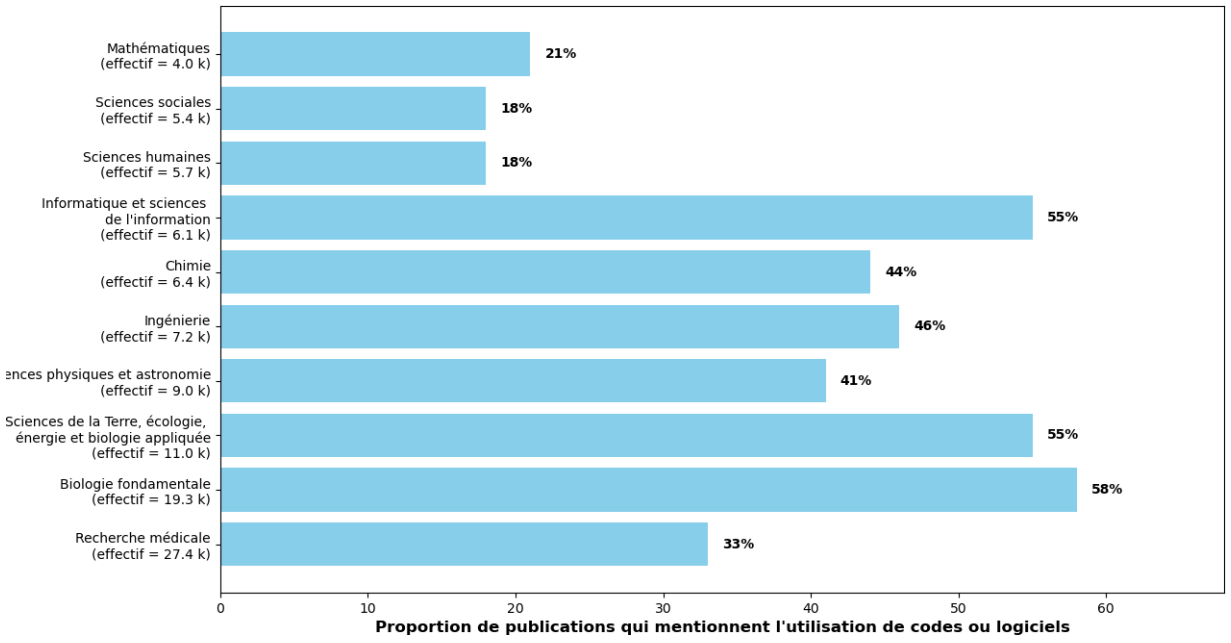
À l'échelle internationale, on constate depuis quelques années une prise de conscience croissante du rôle du logiciel dans la recherche scientifique. Cette prise de conscience se traduit notamment par le déploiement de processus, d'infrastructures et d'outils permettant de recenser, de préserver et de rendre plus visible la production logicielle dans l'écosystème de la science ouverte. Les principes de la science ouverte ne se restreignent pas à l'accès ouvert aux publications scientifiques ; ils s'étendent également aux logiciels, aux données et aux méthodologies, essentiels dans les processus de recherche.

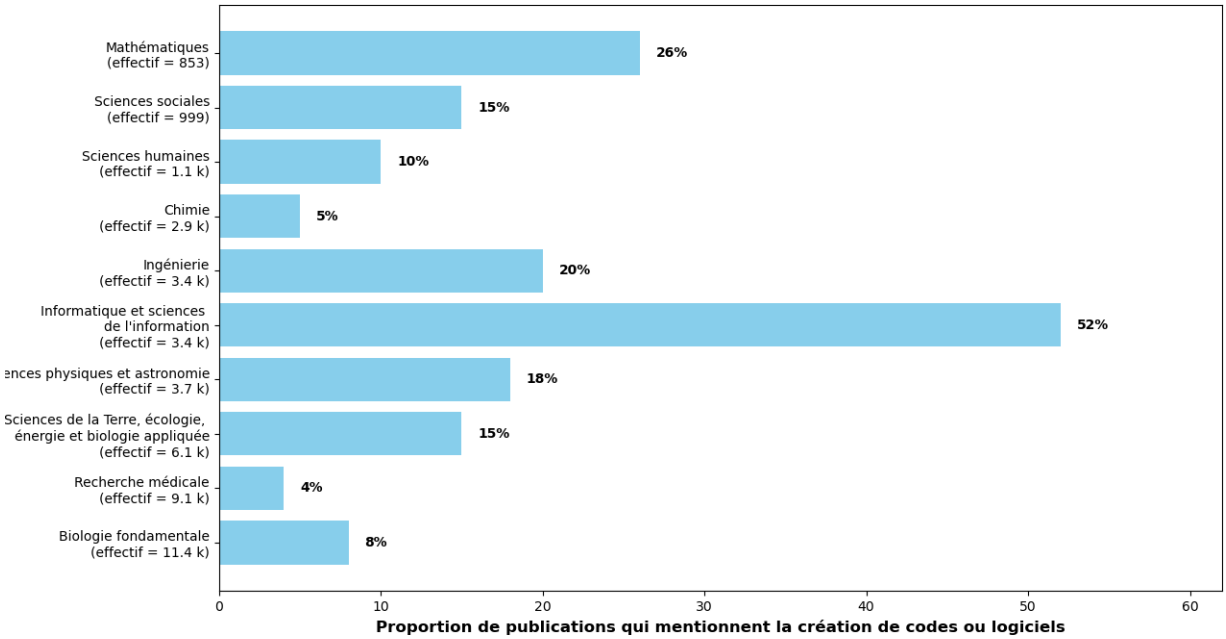
Dans le cadre du présent document, l'expression « logiciels de recherche » désigne des logiciels créés, utilisés ou diffusés dans les productions académiques.

Au sein du collège « Codes sources et logiciels » du Comité pour la science ouverte, le groupe de travail « Identification et mise en avant de la production logicielle de l'enseignement supérieur et de la recherche » a pour premier objectif de réaliser un état des lieux des outils, pratiques et acteurs du domaine. Cet état des lieux constitue une base de travail pour élaborer des préconisations de déploiement d'un catalogue des logiciels développés dans les laboratoires de recherche.

Le logiciel, un sujet transdisciplinaire

L'édition 2023 du baromètre français de la science ouverte indique qu'une proportion significative des publications scientifiques reposent sur des logiciels, et ce, dans toutes les disciplines. Les figures ci-dessous, conçues à partir des données du baromètre de la science ouverte, fournissent un ordre de grandeur et cartographient les domaines académiques concernés.





Ainsi, la production logicielle de l'enseignement supérieur et de la recherche apparaît très nettement comme un sujet transversal.

Le baromètre de la science ouverte s'appuie sur des outils automatisant l'identification de mentions à des logiciels dans des publications. Outre ce repérage effectué sur de larges volumes de données, le rapport sur le logiciel de recherche français, publié le 29 novembre 2023 par le ministère de l'enseignement supérieur et de la recherche, fournit des informations complémentaires. Il synthétise les résultats d'une enquête nationale menée pendant un peu plus d'un an auprès des institutions et personnels de recherche.

Le rapport contient un panorama du type de licences utilisées. Le document ministériel renseigne aussi sur le cadre d'usage des logiciels dans les travaux de recherche et permet de comprendre l'impact obtenu en fonction des différents modèles de valorisation adoptés.

Point de vigilance 1

Les logiciels développés dans le cadre de la recherche académique constituent une production scientifique à part entière. Les logiciels jouent un rôle essentiel dans le processus de recherche dans de très nombreuses disciplines, qu'il s'agisse des STM (sciences, technologies, médecine) comme des sciences humaines et sociales.

Des enjeux désormais mieux identifiés à l'échelle nationale

Dès 2018, l'instance préfiguratrice du collège « Codes sources et logiciels » (le groupe projet « Logiciels libres et open source ») avait analysé les enjeux liés à la production logicielle de l'enseignement supérieur

et de la recherche dans son rapport *Note d'opportunité sur la valorisation des logiciels issus de la recherche* (Clément-Fontaine, Di Cosmo, Guerry, Moreau, et Pellegrini, 2019). Plusieurs des axes décrits dans cette note servent de base au travail restitué dans le présent document.

Un premier enjeu est celui de garantir l'**accès pérenne aux logiciels** créés, utilisés ou diffusés dans les productions de recherche, afin d'en améliorer la reproductibilité et la réfutabilité.

Garantir un accès pérenne suppose de s'appuyer sur une infrastructure telle que l'archive universelle de codes sources Software Heritage. En effet, la durée de vie des plateformes de développement n'est pas garantie sur le long terme, et les changements de politiques d'utilisation de services peuvent également avoir un impact majeur sur les utilisateurs. Une archive logicielle indépendante d'intérêts commerciaux constitue donc une infrastructure de première nécessité.

Fournir un accès pérenne aux logiciels suppose aussi de pouvoir y **faire référence de manière pérenne**.

Autrement dit, les utilisateurs doivent être en mesure de renvoyer vers les versions précises des logiciels concernés, sans risque de lien brisé. Par exemple, un auteur doit pouvoir intégrer un lien perpétuellement valide vers la version X du logiciel Y, telle qu'il l'a utilisée lors de ses travaux.

Il existe de nombreux types d'identifiants répondant aux besoins académiques. Les SoftWare Hash Identifiers (SWHID) sont des identifiants pérennes dédiés aux logiciels. Leur spécification est publique, en cours de normalisation et mise en œuvre dans l'archive Software Heritage. Les SWHID fournissent un service complémentaire aux identifiants tels que les DOI : les besoins sont différents, c'est pourquoi ces solutions ne sont pas redondantes.

L'accès pérenne aux logiciels implique aussi l'identification et la conservation des environnements matériels et des systèmes qui permettent de les ré-**exécuter à l'identique**. Des solutions ont émergé mais ne sont pas encore massivement adoptées. Le sujet de la conservation des environnements n'entre pas dans le périmètre du présent rapport : il s'agit d'un problème scientifique complexe, qui appelle une analyse dédiée.

Un deuxième enjeu est de construire un mécanisme de **citation**. Il s'agit ici de citer directement des logiciels et pas uniquement les articles portant sur des logiciels. Citer directement un logiciel facilite l'identification de la version utilisée plus que ne permet de le faire un article décrivant le logiciel. De plus, comme dans le cas d'autres productions académiques, la citation permet d'attribuer correctement les responsabilités intellectuelles ainsi que de rendre plus visible l'investissement des auteurs et des contributeurs de logiciels. Les efforts pour promouvoir et faciliter la citation de logiciels contribuent à la dynamique de reconnaissance des acteurs du logiciel académique.

Le troisième enjeu est celui de l'accompagnement à la mise en œuvre d'une politique visant à **pérenniser et/ou valoriser de façon adéquate** les productions logicielles des acteurs de la recherche publique. Mettre en œuvre une politique de pérennisation et de valorisation nécessite de s'appuyer sur un **inventaire** permettant d'identifier les productions de manière univoque, c'est-à-dire sans doublons, notamment grâce à des métadonnées complètes et de qualité.

Point de vigilance 2

Le projet de catalogue doit répondre à quatre enjeux principaux pour les logiciels :

• archiver ;

• attribuer des identifiants pérennes adaptés ;

• citer (et faciliter la citabilité) ;

• améliorer la visibilité (et notamment, inventorier l'existant).

Et chacun de ces enjeux suscite des questions d'accompagnement et de services aux chercheurs et chercheuses.

Un état des lieux pour préparer l'action

Un catalogue national devrait être conçu afin de répondre à ces quatre enjeux principaux.

L'objectif du présent rapport est de proposer un socle de recommandations opérationnelles, en définissant, d'une part, ce que recouvrent les quatre enjeux clés et, d'autre part, en analysant les solutions existantes et leurs possibles interactions. Il s'agit tout autant de présenter des outils et des infrastructures que des bonnes pratiques. Cette approche ne se limite donc pas aux éléments techniques.

2 | Définitions

Logiciel de recherche

La notion de « logiciel » recouvre des réalités très différentes. Cette très forte hétérogénéité s'explique par le recours à des pratiques qui varient selon les disciplines et par des contextes de développement technique eux-mêmes pluriels. Le logiciel est par nature un objet profondément protéiforme. Il est donc essentiel d'en préciser la définition.

En recherche, le logiciel peut revêtir différentes formes et peut servir des objectifs variés, comme indiqué dans la *Note d'opportunité sur la valorisation des logiciels issus de la recherche* (Clément-Fontaine, Di Cosmo, Guerry, Moreau, et Pellegrini, 2019). Ainsi, le logiciel peut être : un outil, un résultat de recherche en lui-même, ou un objet de recherche, ces trois catégories n'étant pas exclusives.

Le présent travail s'appuie sur la définition du logiciel de recherche produite par le Collège « Codes sources et logiciels » :

« Les logiciels de recherche sont développés pour répondre à des besoins spécifiques de la science. Ils sont conçus, maintenus, et utilisés par des scientifiques (chercheurs et ingénieurs) et institutions de recherche, éventuellement dans une dimension internationale. Ils peuvent découler de travaux de recherche comme ils peuvent les favoriser, notamment par des publications avant/sur/autour/avec le logiciel. Ceux-ci peuvent se matérialiser de différentes façons (une plateforme, un intergiciel, un workflow ou une bibliothèque, module ou greffon d'un autre logiciel) et être ainsi en interaction dans un écosystème ou au contraire plus autonomes. »

Des contextes de développement pluriels

Un projet de catalogue doit impérativement tenir compte de la complexité intrinsèque du logiciel en recherche afin de traduire l'ensemble de ses facettes et usages :

- les logiciels peuvent être développés par une personne (par exemple un·e doctorant·e, un·e ingénieur·e), une équipe, toute une communauté ;
- les logiciels présentent des niveaux de complexité très différents : cela peut aller de quelques scripts à des programmes composés de millions de lignes. Les niveaux de modularité et de liens à des dépendances externes peuvent aussi être très divers. Cependant, le niveau de complexité architecturale n'augure pas de l'impact du logiciel : un logiciel dont la forme est très simple peut être l'outil majeur d'une communauté ;
- les logiciels ont un état de maturité variable, allant de l'ébauche à usage strictement interne, au logiciel de qualité industrielle utilisé par une vaste communauté, notamment grâce à la fourniture de documentation, de tutoriels, voire de formations ;
- les logiciels ont des durées de vie très diverses, pouvant aller de quelques semaines à des dizaines d'années. Par ailleurs, ils peuvent être dormants pendant quelques mois ou années et être repris ensuite au gré des questions scientifiques qui se posent. Compte tenu de ces spécificités, un logiciel ne peut être comparé à un jeu de données. Ses modalités de conception, de diffusion et de réutilisation soulèvent des problématiques spécifiques.

Point de vigilance 3

Le logiciel est un objet protéiforme, pouvant servir des objectifs très différents. Son cycle de vie ne suit pas forcément une trajectoire linéaire. Sa portée peut être très limitée comme internationale.

Aussi est-il nécessaire de déterminer un ou plusieurs critères sur lesquels fonder le recensement des logiciels. Deux éléments peuvent être pertinents à cet égard :

• l'existence de publications scientifiques associées ; et/ou

• le rôle déterminant d'un logiciel dans un projet de recherche.

Favoriser la réutilisation des logiciels apparaît comme l'un des objectifs prioritaires d'un catalogue, sans pour autant limiter la portée de cet outil à cette finalité. Le recours à de bonnes pratiques de développement détermine très fortement le potentiel de réutilisation des logiciels. Ces bonnes pratiques sont étroitement liées aux facettes de la qualité logicielle, parmi lesquelles on peut citer : la maintenabilité, la présence de documentation, la prise en compte de règles de programmation, les tests, etc. Un travail sur le sujet (David, Colom, Garijo, Castro, Louvet, Ronchieri, Torquati, del Caño, Cerlane, Van den Bossche, Campos, et Di Cosmo, 2024a été réalisé par la Task Force EOSC "Infrastructures for Quality Research Software".

Point de vigilance 4

Le recours à de bonnes pratiques de développement est un point clé pour rendre les logiciels réutilisables. Un catalogue de logiciels peut faciliter la réutilisation, en mettant en lumière les bonnes pratiques mises en œuvre.

Le cas des logiciels « outils » de la recherche

Un certain nombre de logiciels sont développés dans les laboratoires afin de faciliter l'activité de recherche. Ce ne sont pas des productions scientifiques mais ils sont souvent essentiels à la bonne marche des laboratoires et des projets.

Pour ces logiciels génériques et non spécifiques à la recherche elle-même, il existe déjà un outil de référencement : le socle interministériel des logiciels libres.

3 | Analyse des usages et des besoins

Améliorer le potentiel de découverte (selon le principe FAIR : Findable) des logiciels constitue un prérequis à leur réutilisation : « The first step in (re)using software is to find it. Metadata and software should be easy to find for both humans and computers. » (Katz, Gruenpeter, Honeyman, Hwang, Wilkinson, Sochat, Anzt, et Goble, 2021).

Rappel des principes FAIR

Publiés dans Scientific Data en 2016, les principes FAIR (pour Findable, Accessible, Interoperable et Reusable) sont dédiés à la gestion des données de la recherche. Ces principes directeurs fournissent des indications pour les rendre faciles à trouver, accessibles, interopérables et réutilisables par l'homme et la machine.

Pour autant, la recherche d'informations sur les logiciels n'est pas une tâche aisée, y compris pour les utilisateurs les plus experts : « […] even among highly-trained people, the rudimentary approaches of relying on general Web searches, the opinions of colleagues, and the literature were still the most commonly used. » (Hucka et Graham, 2018). Par ailleurs, on note une différence de pratiques entre développeurs et non-développeurs. Ces publics n'utilisent pas les mêmes canaux d'information quand ils cherchent un logiciel prêt à l'emploi (Hucka et Graham, 2018). En amont de toute recommandation sur un catalogue des logiciels de la recherche, il est donc nécessaire de bien identifier et de comprendre les attentes autour d'un tel outil, dédié aux besoins d'utilisateurs finaux variés.

Afin de caractériser ces attendus, une enquête a été menée en mars 2023 auprès d'un large panel, dans le cadre du collège « Codes sources et logiciels ». La synthèse de cette enquête est disponible sur HAL (Granger et Louvet, 2023). Les éléments présentés ci-dessous sont une synthèse de l'analyse des résultats de cette enquête.

Identifier des logiciels via un catalogue : une pratique peu ancrée parmi les répondants

Pour l'utilisateur final, l'intérêt principal d'un catalogue réside dans le fait de pouvoir facilement trouver un logiciel adapté à ses besoins, afin d'éviter des développements redondants. La facilité d'installation et d'exécution, ainsi que la disponibilité du code source, constituent d'autres critères majeurs de sélection d'un logiciel.

Conformément à la littérature existante sur le sujet, il s'avère que, pour les répondants, la recherche de logiciels s'effectue principalement via des outils non spécialisés, tels que des moteurs de recherche généralistes, voire par l'intermédiaire de recommandations des collègues (questions sur des listes de diffusion, échanges informels, etc.). Ce faible recours à des outils spécialisés peut expliquer le scepticisme qui a parfois été exprimé à l'égard d'un catalogue des logiciels de l'enseignement supérieur et de la recherche ; cette opinion est donc à prendre en compte dans l'élaboration de recommandations en la matière.

Le fait que les logiciels fassent rarement l'objet de citations en bonne et due forme peut aussi expliquer pourquoi une partie des utilisateurs finaux ne considère pas un catalogue spécialisé comme un outil de

première nécessité : si la citation de logiciels n'est pas systématique comme l'est celle des articles et des ouvrages, alors le fait de disposer d'un gisement de descriptions standardisées n'apparaît pas comme un enjeu majeur aux yeux de l'utilisateur final.

Avantages d'un catalogue de logiciels

Les bénéfices perçus par les répondants de l'enquête

Selon les répondants, la vocation première d'un catalogue est d'identifier un logiciel répondant à des besoins précis. Ce constat est à mettre en regard du profil des répondants de l'enquête, principalement réparti entre chercheurs et enseignants-chercheurs, d'une part, et personnels d'appui (la plupart ingénieurs), d'autre part.

Point de vigilance 5

Une standardisation des descriptions et une centralisation des données au niveau national faciliteraient la gestion du patrimoine logiciel à l'échelle des individus et des institutions. Ainsi, le projet de catalogue décrit dans le présent document se distingue d'initiatives précédentes, dans la mesure où l'écosystème dédié aux logiciels a lui-même considérablement évolué au cours des dernières années, en France comme à l'étranger. L'enjeu majeur est bien de renforcer l'interopérabilité entre infrastructures afin de mettre à disposition des utilisateurs des procédures de travail automatisées.

Outre les fonctions d'identification, les répondants considèrent qu'un catalogue national pourrait jouer le rôle de tiers de confiance, à la différence d'un moteur de recherche généraliste.

Il ressort également de l'enquête que des descriptions soignées de logiciels pourraient aussi contribuer à encourager les bonnes pratiques de développement et la qualité logicielle. Un tel catalogue, en favorisant l'uniformisation et l'homogénéisation des mentions juridiques, aurait pour conséquence de contribuer à la promotion des logiciels libres.

Par ailleurs, les répondants considèrent que l'identification systématisée des logiciels pourrait en faciliter la conservation : l'archivage du code source des logiciels répertoriés ferait ainsi partie des opérations relatives à l'alimentation du catalogue.

Enfin, le déploiement d'un tel catalogue est perçu par certains des répondants de l'enquête comme l'opportunité de favoriser l'émergence de communautés, de mutualiser les ressources et de collaborer.

La standardisation et l'enrichissement des métadonnées : les perspectives offertes par un catalogue de logiciels

L'enquête préparatoire au projet de catalogue de logiciels a également permis de clarifier les avantages que les répondants associent à ce type d'outil. En complément de ces réponses, d'autres bénéfices et enjeux sont d'ores et déjà identifiés :

- • **gérer des auteurs multiples** : outre les problématiques d'affiliation, se pose la question de l'attribution des responsabilités intellectuelles. Un processus standardisé de description des auteurs multiples peut contribuer à une plus grande clarté, d'autant que les responsabilités intellectuelles des acteurs peuvent évoluer au fil du cycle de vie du logiciel. Un auteur principal peut ainsi devenir un contributeur ;

- **aligner des identifiants** : le recours à des identifiants, c'est-à-dire des codes alphanumériques uniques pérennes, permet d'identifier un auteur ou une structure de façon univoque, dans un environnement numérique donné. L'attribution d'un identifiant permet par exemple de rattacher à un auteur l'ensemble de ses productions, ainsi que de les relier les unes aux autres quelles que soient les sources d'informations qui les décrivent. Pour autant, même dans le case de référentiels interconnectés, des opérations d'alignement resteraient à effectuer pour garantir une interopérabilité satisfaisante. Or, la question du lien entre les différentes productions (publications, codes sources, jeux de données) doit d'ores et déjà être intégrée ;
- **décrire, indexer des logiciels** : le problème n'est pas l'absence de vocabulaires descriptifs pour le logiciel, mais au contraire leur surabondance. La question de l'indexation constitue un enjeu majeur dans le potentiel de découverte des logiciels, et la mise en œuvre d'un catalogue spécialisé apporterait un premier niveau de service aux utilisateurs finaux ;
- **prendre en compte les spécificités du logiciel de recherche** : si les répondants ont souligné la nécessité de fournir des informations sur les mises à jour du logiciel, les différents niveaux de granularité des composants logiciels à décrire doivent être également pris en compte. En effet, la définition du logiciel de recherche apportée par le collège « Codes sources et logiciels » souligne la diversité des types de composants à considérer. Un logiciel peut présenter un grand degré de complexité architecturale ;
- **assurer l'interopérabilité avec le baromètre de la science ouverte** : l'enjeu est de faciliter la mise en avant d'indicateurs.

Structuration et fonctionnalités du catalogue : les attentes des répondants

Les informations souhaitées dans le catalogue portent sur :

- des aspects descriptifs : nom, URL, description ;
- des indicateurs du niveau de dynamisme autour du logiciel : date de la dernière version, etc. ;
- l'existence d'une documentation ;
- les systèmes d'exploitation pris en charge ;
- les conditions de licence.

D'autres éléments ont été mentionnés :

- des statistiques d'utilisation du logiciel : nombre de forks (c'est-à-dire le nombre de copies d'un dépôt de code), de téléchargements, de publications associées, de citations, etc. ;
- le niveau de connaissances techniques et scientifiques requis pour pouvoir utiliser le logiciel ;
- des alternatives ou logiciels similaires.

Outre les besoins classiques d'un moteur de recherche textuel associé à des filtres de recherche, les répondants attendent que l'outil apporte une plus-value par rapport à une simple recherche via un moteur de recherche sous différentes formes :

- la possibilité d'un affichage thématique, permettant en particulier d'identifier facilement les logiciels développés et utilisés dans une communauté ;
- la mise en avant de retours d'utilisateurs, d'avis, de cas d'usages ;
- l'existence d'une interface de programmation applicative (IPA / API) ;
- la mise en avant de logiciels, via des « logiciels à la une ».

Les tableaux suivants synthétisent ces attentes.

Affichage

Nom du logiciel	URL	Description	Mots-clés	Date de la dernière version
Documentation	Licence	Système d'exploitation	Statistiques techniques	Publications
Citations	Alternatives	Niveau d'expertises requis		

Fonctionnalités

Moteur de recherche textuel	Filtres de recherche	Affichage thématique	Retour utilisateurs, avis, cas d'usages	API
Mise en avant de logiciels				

Point de vigilance 6

Un catalogue dédié au logiciel ne constituerait pas un doublon des outils de recherche d'information existants. En effet, les éléments descriptifs à inclure diffèrent de ceux utilisés pour des publications ou des jeux de données ; on peut mentionner par exemple la référence à une maintenance active ou l'existence d'une documentation de qualité. Ces éléments constituent des facteurs déterminants pour les répondants.

Point de vigilance 7

L'enquête a souligné un décalage entre attentes et besoins. En effet, une partie des répondants a indiqué avoir peu d'attentes concernant un catalogue de logiciels de l'enseignement supérieur et de la recherche. Cela ne signifie cependant pas qu'il n'existe pas de besoins.

Les enjeux inhérents à un meilleur recensement des logiciels de recherche ne sont pas identifiés par les utilisateurs finaux potentiels. Cependant, l'enquête a permis de mettre en lumière un socle d'attentes.

4 | Exemples de catalogues

L'objectif de cette section est de faire un état des lieux d'une sélection de catalogues de logiciels de recherche et d'analyser leurs fonctionnalités, en particulier au regard des éléments de la partie précédente.

Les catalogues identifiés ici résultent d'une recherche en ligne et de la sollicitation de certains organismes et communautés. Ils couvrent un périmètre géographique et disciplinaire large. Ils s'adressent à des publics cibles variés et poursuivent des objectifs qui peuvent être assez différents.

Objectifs des catalogues selon le public cible

Publics cibles	Finalités principales du catalogue
Auteurs de logiciels	Fournir un gisement de ressources utilisables ; recenser les logiciels ; rendre les logiciels citables et les citer
Laboratoires et institutions, dont leurs services de valorisation	Constituer une vitrine des productions ; suivre la production scientifique
Public hors enseignement supérieur et recherche	Fournir un gisement de ressources utilisables

L'ensemble des catalogues identifiés et analysés figurent en annexe. On peut noter une grande hétérogénéité en termes de types d'informations affichées et de fonctionnalités, en particulier selon les publics cibles visés. Qui plus est, les informations sur les modalités d'alimentation et de mise à jour des catalogues sont très rarement disponibles sur leurs sites web.

Eléments communs à tous les catalogues : éléments descriptifs et fonctionnalités

La plupart des catalogues étudiés dans le document affichent les informations suivantes :

- le nom du logiciel ;
- son URL (site web ou dépôt sur une forge) ;
- sa description ;
- des mots-clés ;
- l'indication de la licence (qui n'est pas toujours affichée).

En général, on y retrouve également les fonctionnalités suivantes :

- un moteur de recherche textuel ;
- un certain nombre de filtres de recherche ;
- la possibilité d'un affichage thématique.

à périmètre institutionnel

Analyse selon les types de catalogues

Catalogues Il s'agit de catalogues de laboratoires ou d'institutions, dont l'objectif principal est de présenter la production scientifique logicielle de la structure.

En complément des éléments décrits plus haut, on y trouve également souvent l'affichage de la date de la dernière version.

Un exemple de catalogue de ce type est celui proposé par le laboratoire CRIL.

Catalogues à destination des communautés scientifiques

Les catalogues ciblant les communautés scientifiques sont très souvent orientés vers des domaines disciplinaires particuliers.

Ils ont pour objectif de publier une production scientifique logicielle et de faciliter sa réutilisation au sein de la communauté en question. Ces catalogues sont généralement plus détaillés que les autres. En plus des éléments descriptifs généraux, ils fournissent des informations sur :

- la documentation du logiciel (en général, l'URL d'accès) ;
- les systèmes d'exploitation supportés par le logiciel ;
- les métadonnées et mots-clés disciplinaires ;
- les publications scientifiques associées.

Le catalogue SwMath constitue un exemple de catalogue thématique.

Catalogues à destination d'un public plus large que celui de l'enseignement supérieur et de la recherche

Ces catalogues sont d'un usage plus général et s'adressent à un public moins spécialisé. Ils sont conçus pour favoriser la réutilisation des logiciels. De fait, ils proposent souvent des éléments descriptifs facilitant le choix d'un logiciel :

- la date de dernière version ;
- les systèmes d'exploitation supportés par le logiciel ;
- les alternatives disponibles ou les logiciels liés.

Par ailleurs, ces catalogues rendent possibles les interactions à travers des retours d'utilisateurs. Quelques-uns intègrent également des fonctionnalités pour mettre en avant des logiciels.

On peut citer dans cette catégorie le catalogue de l'infrastructure EGI.

Point de vigilance 8

À l'heure actuelle, il n'existe pas vraiment de catalogue qui réponde complètement aux attentes exprimées dans l'enquête : le catalogue doit en effet satisfaire, d'une part, des besoins avancés en recherche d'information scientifique (p. ex., faire le lien entre publication et logiciel) et, d'autre part, inclure des fonctionnalités orientées vers l'utilisation des logiciels (p. ex., gérer des interactions entre utilisateurs).

Questions autour de la gouvernance et des moyens

La mise en place d'un catalogue national de logiciels doit s'accompagner de réflexions concernant la gouvernance à construire mais aussi les moyens nécessaires à sa pérennisation. Un tel catalogue serait en fait une infrastructure, devant être conçue pour le long terme.

Besoins en moyens humains

Ainsi que le souligne le comparatif des fonctionnalités des catalogues, l'offre de services à l'utilisateur peut varier assez fortement d'un catalogue à l'autre.

Les mécanismes de recensement peuvent être très différents : ils peuvent s'appuyer sur une alimentation individuelle du catalogue (p. ex., un auteur décrit son logiciel) ou sur une collecte automatisée (p. ex., les métadonnées d'un gisement d'information sont moissonnées automatiquement). Dans certains cas, des étapes de modération des métadonnées peuvent exister au sein du processus de traitement. Les mécanismes de mise à jour des catalogues (et de leurs sources) devraient être détaillés, car il s'agit d'un point clé pour la pérennité des catalogues. Ces informations sont aussi des indicateurs de fiabilité et de confiance pour l'utilisateur final.

Toutes ces caractéristiques ont un impact important sur les ressources nécessaires à la mise en œuvre, au maintien opérationnel et à l'évolution des catalogues.

Outre les aspects techniques, des compétences en gestion de communautés sont aussi nécessaires au bon fonctionnement de l'ensemble : les services mettant l'accent sur la réutilisation impliquent de créer une dynamique de communauté.

Dans la suite de cette section sera illustrée l'organisation des ressources à mettre en œuvre, en s'appuyant sur deux exemples de catalogues : l'un générique à destination d'un large public, et l'autre disciplinaire.

Code.gouv.fr/public, un catalogue à portée générale

Code.gouv.fr a été créé dans le cadre de la politique de contribution aux logiciels libres de l'Etat, mettant en œuvre l'article 1er de la loi pour une République numérique de 2016 : « Les administrations mentionnées au premier alinéa de l'article L. 300-2 dudit code sont tenues de communiquer […] les documents administratifs qu'elles détiennent aux autres administrations mentionnées au même premier alinéa de l'article L. 300-2 qui en font la demande pour l'accomplissement de leurs missions de service public. ». Or, tout code source obtenu ou développé par un organisme remplissant une mission de service public est considéré comme un document administratif, en vertu de l'article L. 300-2 du code des relations entre le public et l'administration.

Il s'inscrit également dans le deuxième axe du Plan d'action logiciels libres et communs numériques : « Développer et accompagner l'ouverture et la libération des codes sources ».

Code.gouv.fr est une plateforme permettant de réaliser un inventaire des codes sources et des bibliothèques publiés par des administrations, dans le but de :

- faciliter la découverte de ces codes sources ;
- valoriser ces codes sources, notamment ceux fortement susceptibles d'être réutilisés ;
- aider les administrations à mettre en œuvre leur politique de partage et de publication des codes sources ;
- encourager les administrations à réutiliser ou à contribuer aux codes sources publiés ;
- valoriser les contributions des agents à des logiciels libres publiés par l'administration ou à des briques logicielles dont dépendent les codes sources qu'elle publie.

La façon dont cet inventaire est construit est présentée sur cette page. La plateforme s'appuie sur quatre dépôts publiés sous licences libres, qui inventorient des données sur les forges, les organisations et les dépôts publiés par des organismes publics. Le projet data.code.gouv.fr, actuellement en construction, a pour vocation de consolider ces données.

Code.gouv.fr moissonne à ce jour des données issues de **84 forges**, près de **3000 organisations GitHub/GitLab**, et **19000 dépôts**.

Pour naviguer dans cet inventaire très large, la mission « Logiciels Libres » du département « Appui, contrôle et expertise » de la direction interministérielle du numérique a développé Awesome CodeGouvFr. Il s'agit d'une liste de diffusion qui promeut des logiciels libres d'intérêt général produits par les administrations et que tous les organismes publics sont appelés à connaître, voire réutiliser.

Code.gouv.fr a été développé entre 2019 et 2020 par Etalab et a occupé environ un quart d'équivalent temps plein. Sa maintenance prend aujourd'hui peu de temps. Elle est assurée à temps partiel par un personnel de la mission « Logiciels libres ». En 2021, une prestation de 10 jours a été nécessaire pour améliorer codegouvfr-fetch-data. En 2024, une prestation de 30 jours est financée pour déployer data.code.gouv.fr.

swMATH, un catalogue pour la communauté des mathématiques

swMATH fait partie du projet zbMATH Open (anciennement Zentralblatt MATH). Ce projet permet d'accéder facilement à des données bibliographiques, des revues et des résumés dans tous les domaines des mathématiques et de leurs applications, en particulier les sciences naturelles, l'informatique, l'économie et l'ingénierie.

swMath a pour objectif de proposer des informations librement accessibles sur les logiciels mathématiques et fait le lien avec les publications pertinentes associées.

Du point de vue organisationnel, swMATH est géré par le FIZ Karlsruhe. La curation des métadonnées des entrées est gérée par l'ensemble de l'équipe de zbMATH. Par ailleurs, le projet est fortement soutenu par la Société Mathématique Européenne (European Mathematical Society, EMS).

Ce projet repose sur quatre personnes dédiées : deux à temps plein, et deux à temps partiel, correspondant à des profils de chercheur senior, chercheur junior, étudiant stagiaire et administrateur de base de données.

Point de vigilance 9

La question des moyens, et en particulier les ressources humaines, est essentielle à la pérennisation d'un catalogue. Si la phase de gestion de projet ou le déploiement d'une nouvelle fonctionnalité peuvent nécessiter des ressources ponctuelles, celles-ci doivent venir en complément d'une équipe stable, nécessaire pour garantir un bon niveau de service à la communauté. Comme l'a souligné l'enquête, un catalogue scientifique doit en effet répondre à des besoins spécifiques de recherche d'information scientifique, tout en contribuant à la réutilisation des logiciels. Cette double finalité nécessite une bonne compréhension des besoins du public cible.

Un partage d'expériences au niveau international : SciCodes

Il existe de très nombreux catalogues à l'échelle nationale et internationale, mais ces outils sont très hétérogènes et les structures qui les portent ne collaborent pas forcément les unes avec les autres.

Le Consortium SciCodes accueille les éditeurs et les responsables des catalogues et des entrepôts institutionnels de logiciels de recherche. Ses objectifs sont de :

• partager des méthodes de travail, des idées de marketing et des pratiques de communication ;

• présenter les aspects spécifiques des services respectifs de ses membres, discuter des défis et partager des solutions aux problèmes courants qui surviennent dans la gestion des ressources ;

• travailler en collaboration pour accélérer l'adoption des standards CodeMeta et CFF et permettre une meilleure citation, reconnaissance et diffusion des logiciels ;

• travailler en vue d'un standard universel afin de permettre la recherche dans plusieurs registres de logiciels et une interopérabilité entre diverses infrastructures et plateformes.

L'initiative SciCodes a vu le jour dans le but de promouvoir une approche partagée autour du recensement des logiciels. Le consortium a été initialement établi en janvier 2021 par des membres de la task force « Best Practices for Software Registries », formée à partir du groupe de travail FORCE11 SCIWG.

La task force a organisé un atelier en 2019, dont les discussions ont été synthétisées et publiées dans (Monteil, Gonzalez-Beltran, Ioannidis, Allen, Lee, Bandrowski, Wilson, Mecum, Fan Du, Robinson, Garijo, Katz, Long, Milliken, Ménager, Hausman, Spaaks, Fenlon, Vanderbilt, Hwang, Davis, Fenner, Crusoe, Hucka, Wu, Hong, Teuben, Stall, Druskat, Carnevale, et Morrell, 2020). Le consortium SciCodes a un périmètre international, avec une forte représentation d'infrastructures situées aux États-Unis.

Point de vigilance 10 : les leçons du projet PLUME

Le Projet PLUME (Promouvoir les Logiciels Utiles, Maîtrisés et Economiques) et son sous-ensemble RELIER (REférencer les développements Logiciels Internes de l'Enseignement supérieur et de la Recherche) a démarré en 2006 et s'est arrêté en 2013. Son objectif était de partager les expertises internes de l'enseignement supérieur et de la recherche autour des logiciels libres utilisés ou développés dans la communauté, et d'animer cette communauté. Pratiquement, il s'agissait de mettre à disposition des fiches descriptives des logiciels,

précurseur pour le projet actuel de catalogue. Le projet a eu un impact important dans la communauté mais sa pérennité n'a pas pu être assurée, essentiellement pour des questions de gouvernance et de moyens. Des tentatives de faire vivre le projet ont également échoué en particulier en raison du flou concernant les droits d'auteurs sur les notices des logiciels. Ces problématiques sont donc essentielles à anticiper pour toute nouvelle initiative.

5 | Perspectives pour une approche nationale unifiée

Les sections précédentes ont permis d'identifier les attentes et fonctionnalités prioritaires d'un catalogue des logiciels de recherche, ainsi que de présenter des exemples choisis, en regard de ces attendus.

On constate une très grande hétérogénéité des solutions, mobilisatrices de ressources. Des efforts d'harmonisation pourraient être incités, en s'appuyant sur les infrastructures mutualisées existantes.

La présente section recense les infrastructures et initiatives issues de l'environnement de l'enseignement supérieur et de la recherche (ESR) français et international. Elle vise à identifier les manières d'atteindre deux objectifs essentiels au projet de catalogue :

- **utiliser l'existant**, en s'appuyant sur des outils consolidés et ancrés dans les processus de publication et d'information scientifique ;
- **automatiser au maximum** les démarches de recensement et de mise à jour des informations des logiciels au sein du catalogue.

Ces piliers structurels couvrent trois aspects indispensables pour un catalogue de logiciels :

- la notice descriptive ;
- les métadonnées structurées, en portant une attention particulière aux identifiants et aux enjeux de lisibilité par les machines ;
- l'archivage pérenne des sources et leur identification unique.

Software Heritage, l'archive internationale de code source

Software Heritage est une infrastructure portée par Inria et l'Unesco, dédiée à l'archivage du code source des logiciels rendus publics. La bibliothèque a été ouverte en 2015, avec trois missions, qui sont la collecte, la préservation et le partage de code source. Grâce à elle, le code source ainsi que l'historique de développement du logiciel peuvent être préservés et partagés sur le long terme.

Un point d'entrée unifié vers des millions de logiciels

Software Heritage répond à un impératif de centralisation pouvant être envisagé selon deux perspectives :

- la perspective des utilisateurs finaux, qu'ils soient développeurs ou utilisateurs du code source créé par d'autres. L'une des forces de Software Heritage est en effet de centraliser une production logicielle dispersée dans de très nombreux silos. Ce phénomène de dispersion est même constaté à l'échelle d'un projet : un logiciel peut migrer d'une forge à une autre en cours de développement, ce qui peut nuire à sa visibilité et à sa réutilisation.
- la perspective des institutions : si, depuis plusieurs années, les institutions académiques se sont saisies de la problématique des données de recherche, il s'avère que ces solutions ne sont pas adaptées au logiciel. Or, s'appuyer sur une infrastructure mutalisée apparaît comme une nécessité, ainsi que le souligne l'Association des bibliothèques de recherche (ARL) : « No one institution can maintain a collection of software, software environments, and supporting resources (such as expert staff or specialized technology) sufficient to facilitate access to all of the digital resources that may be of interest to its researchers. » (The Association of Research Libraries, 2018).

Les modalités d'alimentation de l'archive

Les fonds de Software Heritage sont constitués au moyen de quatre processus distincts :

- le moissonnage : ce processus automatisé permet de collecter de larges volumes de code source. Sont moissonnées des plateformes telles que GitHub, CRAN, Python Package Index, etc. La sélection des contenus à archiver automatiquement s'effectue à l'échelle des forges ou des plateformes de distribution, et non à l'échelle de chaque projet logiciel. Si Software Heritage préserve des logiciels de nature patrimoniale, les fonds de la bibliothèque ne se limitent pas à du code source ancien, bien au contraire ;
- le dépôt manuel : la fonctionnalité « Save code now » permet à chacun d'archiver ou de mettre à jour l'état courant d'un logiciel, indépendamment de la fréquence standard du moissonneur. Cette fonctionnalité se décline en différentes options, plus ou moins automatisables, et peut être utilisée par tous : il n'est en effet pas nécessaire d'être l'auteur du code source pour effectuer un « Save code now ».
- l'archivage d'une forge à la demande : si la fonction « Save code now » permet l'archivage personnalisé d'un projet logiciel, le service « Add forge now » permet de le faire à l'échelle d'une forge. Si le moissonneur est déjà configuré pour collecter les données issues des forges centralisées, les instances locales doivent faire l'objet d'un processus « Add forge now ».
- le dépôt à la demande dans le cadre d'un partenariat : au moyen du protocole SWORD, des logiciels clients mis en œuvre par les entités partenaires transfèrent leurs données et fichiers à Software Heritage. C'est le cas pour le CCSD (HAL, voir section suivante), les journaux scientifiques eLife, IPOL, etc.

Software Heritage applique un processus de dédoublonnage : ainsi, un fichier présent dans différents projets est stocké une seule fois et référencé par différents nœuds du graphe des projets. En mai 2024, 19 milliards de fichiers de code source uniques ont ainsi été archivés.

Une infrastructure à vocation internationale solidement ancrée dans la politique française de science ouverte

L'infrastructure Software Heritage est intégrée dans les politiques de science ouverte de plusieurs établissements, ainsi que dans le plan national pour la science ouverte, et est referencée dans l'édition 2022 de la feuille de route nationale des infrastructures de recherche.

Par ailleurs, Software Heritage a mis en place un programme d'ambassadeurs et propose de nombreuses ressources pour l'accompagnement des utilisateurs.

HAL, l'archive ouverte pluridisciplinaire nationale

HAL est l'archive nationale choisie par la communauté scientifique et universitaire française pour la diffusion ouverte de ses résultats de recherche. Multidisciplinaire, elle permet de partager en libre accès les résultats de recherche, publiés ou non. Elle est au service des chercheur.es affilié.es à des institutions académiques, qu'elles soient publiques ou privées.

HAL est aussi une plateforme mutualisée qui rassemble plus de 140 archives institutionnelles d'universités, de grandes écoles et d'organismes de recherche.

Elle couvre de nombreux types de documents produits par la recherche : articles publiés ou non dans des revues à comité de lecture, communications scientifiques, thèses, preprints, logiciels, etc. En mars 2024, elle comprenait plus de 1,3 millions de documents avec texte intégral.

Créée en 2001 à l'initiative du physicien Franck Laloë, elle est développée par le Centre pour la communication scientifique directe (CCSD), unité d'appui et de recherche sous la tutelle de trois organismes de recherche publics : CNRS, Inria et INRAE.

Le CCSD assure l'administration de l'infrastructure matérielle, les développements logiciels et la maintenance applicative. Il dispose d'un service de support pour répondre aux demandes des utilisateurs, et les accompagne en organisant des formations. Le CCSD procède régulièrement à des alignements avec d'autres référentiels (ORCID, ROR, IdRef et RNSR) et à la curation des données.

Les principes fondateurs de HAL

HAL se fonde sur un ensemble de principes qui garantissent l'accès et la préservation à long terme des travaux scientifiques qu'elle diffuse :

- une infrastructure OAI garantissant l'interopérabilité ;
- la stabilité des identifiants (URL des dépôts, notamment) ;
- la préservation à long terme des documents ;
- la qualité scientifique des documents déposés ainsi que des métadonnées qui les décrivent ;
- l'horodatage des dépôts, qui établit la paternité du document déposé ;
- une interconnexion aux grands serveurs internationaux (arXiv, REPEC, SWH) et aux principaux référentiels et sources de métadonnées (ORCID, RNSR, ROR, OpenAire, data.gouv.fr, Crossref et Datacite) ;
- des services pour valoriser (identifiant chercheur idHAL, CV, collection, portails), piloter (statistiques), consulter et exploiter les contenus (moteur de recherche, API, Triplestore, TEI, etc.) ;
- l'export dans différents formats standardisés (Bibtex, DC, TEI, Endnote) ;
- l'import TEI via le protocole standardisé SWORD ;
- le référencement renforcé des données de la plateforme, grâce à une collaboration avec Google Scholar et avec de nombreuses plateformes (OpenAIRE, OpenAlex, Isidore, etc.).

HAL constitue ainsi un instrument ouvert et pérenne de souveraineté scientifique.

Le dépôt de codes sources dans HAL : une fonctionnalité issue d'une collaboration

Une collaboration entre le CCSD et Software Heritage (Di Cosmo, Gruenpeter, et Zacchiroli, 2020) a permis d'élargir le dépôt dans HAL aux logiciels et codes source.

Le dépôt dans HAL permet de favoriser la citabilité et la valorisation de ce type de production scientifique, l'archivage étant pris en charge par l'infrastructure Software Heritage. Afin d'améliorer la description des logiciels, HAL intègre des métadonnées spécifiques au logiciel (langage de programmation, dépôt de code, système d'exploitation, version, état du développement, outils de développement, identifiant SWHID), basées sur le schéma CodeMeta.

Outre l'adaptation technique de HAL aux spécificités du logiciel, le CCSD propose une formation dédiée à la modération des dépôts logiciels, en partenariat avec Software Heritage. Enfin, le processus de dépôt dans HAL fait d'ores et déjà l'objet d'une documentation pour les déposants comme pour les modérateurs.

HAL et Software Heritage proposent ainsi des services complémentaires au bénéfice de la dissémination scientifique et de la reproductibilité des résultats.

L'inscription de HAL dans la politique nationale de science ouverte

L'archive HAL est soutenue par les deux Plans Nationaux pour la Science Ouverte et est labellisée infrastructure de recherche par le ministère de l'enseignement supérieur et de la recherche au titre de HAL+ (qui repose sur HAL), Episciences et Sciencesconf. Elle constitue un axe majeur des politiques institutionnelles pour l'ouverture des publications.

Le CCSD a également mis en place ln programme Ambassadeurs et Ambassadrices HAL pour renforcer le dialogue avec les communautés scientifiques.

Point de vigilance 11

En France, des infrastructures ont vocation à jouer un rôle clé dans l'élaboration d'un catalogue logiciels : HAL et Software Heritage.

Ces infrastructures collaborent déjà ensemble et sont intégrées à la feuille de route nationale pour le développement de la science ouverte. Le projet d'un catalogue national s'inscrit pleinement dans cet écosystème.

Normaliser l'échange de métadonnées logicielles : des initiatives internationales

Le rapport sur les infrastructures académiques pour les logiciels publié par l'EOSC en 2020 European Commission, 2020) souligne trois enjeux relatifs aux métadonnées :

- développer des formats lisibles par les machines et renforcer la standardisation ;
- décrire de manière adaptée les rôles des auteurs et des contributeurs de logiciels ;
- lier les unes avec les autres les différentes productions académiques.

Dans ce but, on considère deux grandes catégories de métadonnées :

- métadonnées intrinsèques : elles fournissent des informations essentielles pour assurer la préservation et l'utilisabilité à long terme des logiciels. Elles sont contenues dans un fichier à vocation descriptive, incorporé aux fichiers qui composent le code source du logiciel. C'est pourquoi, très souvent, les auteurs de logiciels sont les premiers et les principaux fournisseurs de métadonnées intrinsèques ;
- les métadonnées extrinsèques : elles fournissent quant à elles des informations importantes sur le contexte et la provenance du logiciel, telles que la forge sur laquelle le logiciel a été développé. Ces informations aident les utilisateurs à localiser le logiciel, à le comprendre, à identifier la communauté qui l'utilise. Elles renseignent également sur la relation entre le logiciel et d'autres produits de recherche tels que des publications et des jeux de données. Ces métadonnées sont qualifiées d'extrinsèques car elles ne sont pas incluses dans le code source. Elles peuvent être attribuées par des services éditoriaux, des agrégateurs, des archives institutionnelles, etc. Dans ce cas, le code source du logiciel et les métadonnées extrinsèques doivent chacun faire l'objet d'un archivage.

Deux éléments principaux sont à souligner : d'une part, métadonnées intrinsèques et extrinsèques ont des rôles distincts dans la gestion, la conservation, la préservation et l'utilisation des métadonnées. D'autre part, les auteurs de logiciels jouent un rôle majeur dans la description de leurs travaux.

Il faut cependant noter que la description des logiciels est une pratique nettement moins encadrée que celle des ouvrages ou des articles. La difficulté réside dans le foisonnement des vocabulaires disponibles, tels que SoftwareX Code metadata ou Edam. Cet aspect doit être pris en compte afin de guider efficacement les auteurs de logiciels, ainsi que les autres acteurs qui interviennent sur les métadonnées. Toutefois, des préconisations émergent à l'échelle européenne (Gruenpeter, 2023) et permettent d'identifier des niveaux de priorité au sein de chaque catégorie de métadonnées.

CodeMeta, un format d'échange et une palette d'outils pour créer des métadonnées intrinsèques

La finalité de CodeMeta est d'améliorer la description des logiciels de recherche en faisant le lien entre les différents vocabulaires.

Codemeta est une initiative portée par des acteurs du monde de la recherche, de l'ingénierie et de l'éducation. Il s'agit de prolonger les travaux réalisés dans le cadre de « Code as a Research Object », un projet du Mozilla Science Lab dont l'objectif était d'améliorer l'intégration du logiciel et du code source dans les processus de travail académiques. Le comité de gestion du projet actuel réunit des représentants des institutions suivantes : Caltech Library, GitHub, Mozilla Science Lab, Software Heritage, UC Berkeley et Universidad Politécnica de Madrid.

Les auteurs de logiciels interviennent très en amont du processus de description car ils fournissent les métadonnées intrinsèques. De fait, préconiser l'utilisation de CodeMeta peut faciliter leur pratique.

CodeMeta est un format d'échange, exprimé en JSON-LD, pour les métadonnées logicielles. Outre les informations générales sur un logiciel (nom, URL, date de création, date de mise à disposition, etc.), CodeMeta permet de fournir des informations techniques, juridiques ainsi que des éléments sur le processus de développement lui-même (renvoi vers un service d'intégration continue, vers un outil de suivi de bogues, etc.). Un groupe de métadonnées est dédié à la citabilité du logiciel et à son potentiel de découvrabilité : mots clés, domaine disciplinaire, financement, identifiant unique peuvent ainsi être complétés. Enfin, CodeMeta offre la possibilité de décrire de manière détaillée les différents types de responsabilités intellectuelles et de contribution.

Rappelons que métadonnées intrinsèques et extrinsèques sont mutuellement complémentaires et sont diffusées, exploitées et conservées de manières différentes.

L'objectif et la valeur ajoutée de CodeMeta par rapport à d'autres formats est de créer un vocabulaire conceptuel qui peut être utilisé pour normaliser l'échange de métadonnées logicielles entre les référentiels et les organisations.

La force de CodeMeta est d'étendre le vocabulaire de Schema.org, un schéma de métadonnées généraliste conçu dans une perspective d'amélioration de l'indexation des données par les moteurs de recherche. Le format CodeMeta est ainsi particulièrement adapté aux besoins académiques, et peut être exploité par des traitements automatisés.

CodeMeta joue le rôle de pivot entre les différents vocabulaires disponibles : « CodeMeta provides a common vocabulary and crosswalks to software metadata concepts from 25 common software metadata dialects, which provides the ability to automate the conversion of software among these 25 dialects. » (Katz, Bouquin, Hong, Hausman, Chivvis, Clark, Crosas, Druskat, Fenner, Gonzalez-Beltran, Gruenpeter, Habermann, Haines, Harrison, Henneken, Hwang, Jones, Kelly, Kennedy, Leinweber, Rios, Robinson, Todorov, Wu, et Zhang, 2019).

Une palette de services pour faciliter la description des logiciels

Si CodeMeta présente de nombreux avantages techniques, c'est aussi un format autour duquel se sont déployés des services et une communauté. Ainsi, le CodeMeta generator, conçu pendant le hackathon Force2019, permet à ses utilisateurs de vérifier si le fichier produit est correctement normé. Il existe aussi des outils permettant d'alimenter automatiquement un fichier CodeMeta :

- paquetage codemetaR, pour les utilisateurs du langage R ;
- l'outil CodeMetaPy, pour les utilisateurs de Python.

Adoption de CodeMeta par les infrastructures de référence en France

Software Heritage et HAL exploitent déjà les métadonnées issues des fichiers CodeMeta. Par exemple, lorsqu'un auteur dépose un logiciel dans HAL en copiant-collant le SoftWare Hash Identifier (SWHID) obtenu depuis l'interface de Software Heritage, les métadonnées issues du fichier CodeMeta pré-remplissent les champs descriptifs de HAL. Le travail de saisie de l'auteur est ainsi considérablement réduit. HAL offre également la possibilité d'ajouter des métadonnées extrinsèques. Ces procédures permettent de combiner les deux catégories de métadonnées.

Enfin, CodeMeta repose sur une communauté croissante d'utilisateurs qui contribuent activement au développement de ce format, dont la version 3 a été diffusée en juillet 2023.

Schema.org, l'initiative historique

Comme il a été mentionné, CodeMeta est dérivé de Schema.org ; il consiste en effet en une extension des classes « SoftwareApplication » et « SoftwareSourceCode » de ce dernier.

Schema.org est une initiative collaborative (« open community process ») fondée par Google, Yahoo, Yandex et Microsoft. Son but est de promouvoir des schémas de données pour le web afin d'améliorer l'indexation des données par les moteurs de recherche. Schema.org est donc un standard de fait, à la différence des normes débattues au sein du W3C. Les schémas sont un ensemble de « types ». Chaque « type » est associé à des propriétés et organisé de manière hiérarchisée. Ainsi, le logiciel relève du type « creative work ».

Schema.org reste évolutif, pour répondre à la nécessité de s'adapter le plus possible aux objets décrits, afin que les moteurs de recherche puissent facilement les référencer.

Au-delà de CodeMeta : Citation File Format (CFF), un format descriptif très répandu

Il existe de nombreuses initiatives visant à une meilleure description des logiciels. L'enjeu de cette section est moins de dresser un état des lieux que de fournir un aperçu du type de choix auxquels les auteurs de logiciels peuvent être confrontés, en prenant l'exemple de CFF.

CFF est un format très utilisé par les communautés de développeurs, en raison de son intégration à GitHub, l'une des forges les plus populaires depuis plusieurs années. On peut noter que les fichiers CFF, à l'instar des fichiers CodeMeta, sont indexés par Software Heritage. CFF et CodeMeta ne s'excluent pas. Il est donc possible pour les auteurs de combiner les formats CFF et CodeMeta dans un même processus descriptif : « *While CFF is a suitable format for the initial provision of software citation metadata by the creators of a software, the metadata it provides should also be transferred to CodeMeta downstream in the software citation workflow. »

Les fichiers CFF sont des fichiers plein texte, lisibles par l'humain et les machines. Mais CFF est un format plus limité que CodeMeta. (Katz, Bouquin, Hong, Hausman, Chivvis, Clark, Crosas, Druskat, Fenner, Gonzalez-Beltran, Gruenpeter, Habermann, Haines, Harrison, Henneken, Hwang, Jones, Kelly, Kennedy, Leinweber, Rios, Robinson, Todorov, Wu, et Zhang, 2019).

Autre élément d'intérêt : les systèmes de badges

Les métadonnées, qu'elles soient intrinsèques ou extrinsèques, fournissent des informations essentielles pour la réutilisation d'un logiciel. Cependant, ces métadonnées n'ont pas valeur de test. Or, les enjeux de reproductibilité de la recherche constituent un défi majeur dans le contexte de la science ouverte. C'est pourquoi des communautés ont proposé et développé des systèmes de badges. Leur objectif n'est pas d'évaluer la qualité d'un logiciel, c'est-à-dire de définir si le logiciel répond au problème posé, mais d'attester que le logiciel fonctionne comme il est supposé le faire. Le plus ancien système de badges a été lancé dès 2011.

Les évaluateurs peuvent attribuer différents badges à un artefact, en fonction de ses diverses qualités (disponible, reproduit, voire répliqué). L'Association for Computer Machinery fournit une explication complète des badges concernés.

Ces systèmes de vérification tendent à se développer, du fait des éléments de confiance qu'ils apportent aux productions scientifiques. Des initiatives plus récentes ont ainsi vu le jour, comme celle de la National Information Standards Organization.

De façon concomitante, certaines communautés d'informaticiens (notamment en méthodes formelles, ingénierie logicielle et sécurité logicielle) ont proposé d'ajouter au système de relecture d'article des

processus concernant l'évaluation d'artefacts. Un artefact consiste en un instantané de l'environnement informatique et du logiciel utilisé pour l'évaluation expérimentale de l'article (par exemple, un conteneur Docker ou une image de machine virtuelle).

Dans le cadre de ces processus d'évaluation composites, les auteurs sont encouragés à soumettre un artefact logiciel en plus de leur article. Ces artefacts sont construits et évalués pour s'assurer que toutes les affirmations expérimentales liées au logiciel utilisé dans l'article puissent être reproduites par un tiers. La réussite à ces tests peut être matérialisée par l'obtention des badges correspondants.

6 | Une dynamique inter-métiers autour de l'accompagnement

L'élaboration d'un catalogue de logiciels fait émerger des questions de nature technique autant qu'organisationnelle. En effet, la collaboration et la mutualisation à l'échelle d'un établissement ou d'une institution constituent des enjeux majeurs pour mener à bien ce type de projet.

Un périmètre élargi d'acteurs à faire collaborer ensemble

L'un des enjeux d'un catalogue de logiciels est de centraliser des informations produites et gérées par une grande diversité d'acteurs n'ayant pas forcément l'habitude de collaborer ensemble et ayant une utilisation différente de ces données.

On peut ainsi citer :

- personnels d'appui dans les unités de recherche ;
- les directions de la recherche et de la valorisation ;
- les directions des systèmes d'information ;
- les services de documentation et de bibliothèques ;
- les services dédiés à la valorisation et au transfert.

Par ailleurs, l'identification des parties prenantes peut s'avérer délicate, dans la mesure où certaines missions sont associées à des profils en cours d'émergence, parfois difficiles à définir (Jetten, Grootveld, Mordant, Jansen, Bloemers, Miedema, et van Gelder, 2021) :

« *As data stewardship and research software engineering are new fields of expertise, with a mixture of existing and newly to be recruited professionals with various backgrounds and expertise, it is challenging to capture this in a job profile. Even if good practices and example job profiles exist, most organisations are not aware of these examples, as they are not shared among organisations* ».

De ce point de vue, les travaux menés à l'étranger par les structures citées ci-dessous peuvent nourrir la réflexion :

- Society of Research Software Engineering ;
- ReSA, Research Software Alliance ;
- FAIR for Research Software, Working Group de la Research Data Alliance.

Enfin, aux acteurs locaux s'ajoutent des réseaux métiers nationaux. L'ensemble des acteurs et des outils clés à l'échelle de l'écosystème académique est résumé dans le schéma ci-dessous(WG, Allen, Bandrowski, Chan, Di Cosmo, Fenner, Garcia, Gruenpeter, Jones, Katz, Kunze, Schubotz, et Todorov, 2020) :

Point de vigilance 12

Le déploiement d'un catalogue de logiciels nécessite des ressources humaines dédiées ainsi qu'une réflexion anticipée sur l'accompagnement des acteurs impliqués dans la gestion du catalogue. De plus, ces différentes communautés d'acteurs sont amenées à jouer un rôle clé dans l'appropriation du catalogue par les communautés scientifiques.

Assurer la qualité des métadonnées, un enjeu premier

Assurer la qualité des métadonnées implique de former les personnels dédiés aux spécificités du logiciel et de sensibiliser les auteurs de logiciels à certaines bonnes pratiques (p. ex., ajouter un fichier de type CodeMeta.json au code source, afin de fournir des métadonnées intrinsèques exploitables automatiquement).

Afin de développer un socle unifié de compétences, les organismes de formation des personnels ont un rôle majeur à jouer. Il est possible de s'appuyer sur l'expertise déjà développée par des services tels que les centres régionaux de formation aux carrières des bibliothèques, ou les unités régionales de formation à l'information scientifique et technique. Des structures nationales, telles que l'école nationale supérieure des sciences de l'information et des bibliothèques (ENSSIB) et le CCSD, proposent une offre de formation adaptée aux professionnels de l'information scientifique, dans la mesure où les spécificités de l'objet logiciel sont bien intégrées dans les formations proposées.

Des collaborations avec les services de formation des personnels des EPST et des universités peuvent quant à elles permettre de toucher les autres catégories de personnels de soutien à la recherche.

S’appuyer sur une dynamique nationale, à l’instar de ce qui est fait pour les données de recherche

Au niveau national, une structuration est en cours concernant l’accompagnement à la gestion des données, avec le soutien de relais locaux. Les ateliers de la donnée constituent un exemple de cette structuration articulant un niveau local avec un niveau national. Ils peuvent jouer un double rôle :

- contribuer à former les utilisateurs finaux aux enjeux descriptifs relatifs au logiciel de recherche ;
- contribuer à former les personnels de soutien à la recherche.

Le contexte actuel français est favorable à un développement similaire, pour le logiciel, à ce qui se construit autour des données.

Certains ateliers de la donnée ont déjà intégré la problématique du logiciel à leur offre de formation : on peut citer l’exemple de Data Univ Eiffel, l’initiative de l’université de Lille à l’occasion du Printemps de la donnée, ainsi que celle de l’Université Grenoble Alpes, qui propose des formations récurrentes pour les doctorants ainsi que des webinaires pour un public plus large, mais aussi un accompagnement spécifique concernant les problématiques autour du logiciel.

Au sein des ateliers de la donnée, un groupe de travail dédié au logiciel existe déjà. L’une de ses priorités est de familiariser les professionnels de l’information scientifique aux spécificités du logiciel.

Une offre de services pouvant appeler des compétences très différentes

Au-delà d’un socle de base de compétences pour assurer et maintenir la qualité des métadonnées, il s’avère que les compétences à mobiliser par les personnels de soutien à la recherche vont dépendre du type de services associés au catalogue.

En effet, la comparaison des catalogues sélectionnés souligne une diversité des modalités d’alimentation lorsque le périmètre des logiciels à identifier est plus large que celui d’une seule entité (laboratoire ou institution).

Certains catalogues s’appuient sur des processus automatisés de collecte de données (p. ex., swMATH), alors que d’autres résultent d’une contribution manuelle et volontaire (p. ex., ASCL, NITRC, etc.). Lorsque l’accent est mis sur la contribution des utilisateurs finaux (qu’ils soient auteurs de logiciels ou utilisateurs), on constate très souvent la mise en place d’un écosystème d’accompagnement de la communauté, incluant l’animation de communauté de pratique, des formations et manifestations, etc.

Or, ce type d’accompagnement diffère de celui qui peut être mis en place à destination de professionnels de l’information fonctionnant de manière distribuée, comme c’est le cas par exemple pour le catalogue du Sudoc. Ces professionnels bénéficient d’un accompagnement et d’une coordination nationale assurés par l’Abes. Cependant, ce type de services n’est pas transposable à toute communauté (chercheurs, ingénieurs logiciels de recherche, etc.).

Dans les deux cas toutefois, on note que le déploiement d’un catalogue alimenté de manière distribuée nécessite des moyens dédiés à des fonctions d’accompagnement, en plus de ceux dédiés aux aspects techniques (infrastructure, services en ligne, etc.).

7 | Conclusions et perspectives

Le logiciel est un élément essentiel des processus de recherche dans la quasi-totalité des disciplines scientifiques, concernant tant les sciences dites dures que les sciences humaines et sociales. L'approche peut être différente d'une communauté à l'autre, selon que le logiciel y est considéré comme un outil, un résultat ou un objet d'études, mais son rôle reste primordial.

Le présent état de lieux permet de formaliser le besoin d'un catalogue pour l'identification et la mise en avant des logiciels de l'enseignement supérieur et de la recherche. Un tel catalogue vise à répondre à plusieurs problématiques, telles que la nécessité d'apporter de la visibilité, de préserver le code source, de citer directement un logiciel plutôt que l'article qui le décrit, de faire le lien avec les publications et les données etc.

Afin d'apporter les premiers éléments nécessaires à la construction d'un catalogue des logiciels de l'enseignement supérieur et de la recherche, une enquête nationale a permis d'identifier des attentes prioritaires. C'est à la lumière de ces éléments qu'a été réalisée l'analyse d'une sélection d'initiatives et d'infrastructures existantes à l'échelle nationale et internationale. Le choix s'est porté sur des infrastructures jouant déjà un rôle majeur. L'analyse a également été guidée par les travaux menés à l'échelle européenne.

Depuis les premiers catalogues de logiciels, l'écosystème s'est considérablement enrichi et diversifié.

D'une part, ce rapport permet de comprendre comment formats et infrastructures peuvent s'articuler afin d'automatiser au maximum les processus. D'autre part, il souligne le besoin d'inclure dès le début de la réflexion les questions d'accompagnement des différents acteurs du logiciel, qu'il s'agisse des auteurs de logiciels ou des personnels de soutien à la recherche. En effet, la multiplicité des outils et la faible standardisation des pratiques ajoutent un niveau de complexité indéniable. Pour autant, la dynamique impulsée autour des données de recherche offre un cadre propice pour développer des actions de soutien.

Point de vigilance 13

Depuis quelques années, à l'échelle nationale et internationale, le logiciel est devenu l'objet d'une plus forte attention dans le milieu de la recherche. Un foisonnement d'activités et d'outils sur le sujet en découle. Si l'on peut se réjouir de cette reconnaissance progressive du logiciel en tant que réalisation académique, il ne faut pas que la diversité des actions conduise à une dispersion des énergies. De tels projets nécessitent des ressources pérennes pour mettre en œuvre leurs aspects techniques et fournir un support aux utilisateurs. De plus, une approche de capitalisation et de mutualisation apparaît aussi comme un facteur clé pour l'adoption de nouvelles pratiques : l'enjeu est de proposer aux utilisateurs finaux un ensemble de services performants car intégrés à leurs pratiques.

Point de vigilance 14

Ce premier livrable constitue un socle préalable à la proposition de préconisations techniques, organisationnelles et de gouvernance, concernant un futur catalogue national des logiciels de recherche.

Bibliographie indicative

Clément-Fontaine M., Di Cosmo R., Guerry B., Moreau P., Pellegrini F. « Note d'opportunité sur la valorisation des logiciels issus de la recherche ». Comité pour la science ouverte [En ligne]. 2019. Disponible sur : https://doi.org/10.52949/17

David M., Colom M., Garijo D., Castro L. J., Louvet V., Ronchieri E., Torquati M., Del Caño L., Cerlane L., Van Den Bossche M., Campos I., Di Cosmo R. « Ensure Software Quality ». 2024. Disponible sur : https://doi.org/10.5281/ZENODO.10723608

Di Cosmo R., Gruenpeter M., Zacchiroli S. « Referencing Source Code Artifacts: A Separate Concern in Software Citation ». Computing in Science & Engineering [En ligne]. 2020. Disponible sur : https://doi.org/10.1109/MCSE.2019.2963148

European Commission Directorate-General For Research & Innovation. Scholarly infrastructures for research software – Report from the EOSC Executive Board Working Group (WG) Architecture Task Force (TF) SIRS [En ligne]. [s.l.] : Publications Office, 2020. Disponible sur : https://doi.org/doi/10.2777/28598

Granger S., Louvet V. Rapport sur l'enquête sur les usages d'un catalogue logiciels [En ligne]. [s.l.] : HAL, 2023. Disponible sur : https://hal.science/hal-04829203

Gruenpeter M. Developing Guidelines for Metadata Collection and Curation for Research Software [En ligne]. 2023. Disponible sur : https://doi.org/10.5281/zenodo.7962734

Hucka M., Graham M. J. « Software search is not a science, even among scientists: A survey of how scientists and engineers find software ». Journal of Systems and Software [En ligne]. 2018. Disponible sur : https://doi.org/10.1016/j.jss.2018.03.047

Jetten M., Grootveld M., Mordant A., Jansen M., Bloemers M., Miedema M., Van Gelder C. W. G. « Professionalising data stewardship in the Netherlands. Competences, training and education. Dutch roadmap towards national implementation of FAIR data stewardship. » 2021. Disponible sur : https://doi.org/10.5281/zenodo.4623713

Katz D. S., Bouquin D., Hong N. P. C., Hausman J., Chivvis D., Clark T., Crosas M., Druskat S., Fenner M., Gonzalez-Beltran A., Gruenpeter M., Habermann T., Haines R., Harrison M., Henneken E., Hwang L., Jones M. B., Kelly A. A., Kennedy N., Leinweber K., Rios F., Robinson C. B., Todorov I., Wu M., Zhang Q. « Software Citation Implementation Challenges ». 2019. p. 26.

Katz D. S., Gruenpeter M., Honeyman T., Hwang L., Wilkinson M. D., Sochat V., Anzt H., Goble C. « A Fresh Look at FAIR for Research Software ». FAIR for Research Software (FAIR4RS) subgroup [En ligne]. 2021. Disponible sur : https://doi.org/10.48550/arXiv.2101.10883

Monteil A., Gonzalez-Beltran A., Ioannidis A., Allen A., Lee A., Bandrowski A., Wilson B. E., Mecum B., Fan Du C., Robinson C., Garijo D., Katz D. S., Long D., Milliken G., Ménager H., Hausman J., Spaaks J. H., Fenlon K., Vanderbilt K., Hwang L., Davis L., Fenner M., Crusoe M. R., Hucka M., Wu M., Hong N. P. C., Teuben P., Stall S., Druskat S., Carnevale T., Morrell T. Nine Best Practices for Research Software Registries and Repositories: A Concise Guide [En ligne]. 2020. Disponible sur : https://arxiv.org/abs/2012.13117

WG Research Data Alliance/FORCE11 Software Source Code Identification, Allen A., Bandrowski A., Chan P., Di Cosmo R., Fenner M., Garcia L., Gruenpeter M., Jones C. M., Katz D. S., Kunze J., Schubotz M., Todorov I. T. Software Source Code Identification Use cases and identifier schemes for persistent software source code identification [En ligne]. décembre 2020. Disponible sur : https://doi.org/10.15497/RDA00053

Remerciements

Merci à toutes les personnes listées ici qui ont apporté une contribution à ce rapport, sous différentes formes :

- Maxence Azzouz-Thuderoz
- Florent Chuffart
- Bastien Guerry
- Hélène Jonin
- Daniel Le Berre
- Grégory Miura
- Raphaël Monat
- Samuel Thibault

Merci également aux personnes qui ont relu et permis d'enrichir le document :

- Hélène Bégnis
- Arnaud Bey
- Roxanne Hablot
- Alexis Lebis
- Sébastien Mazzarese
- Jozefina Sadowska
- Maria-Grazia Santangelo

A | Liste des catalogues analysés

Catalogue	Description	Type de catalogue	Remarques
https://code.gouv.fr/public/	Codes sources du secteur public	Public	Référence les dépôts publiés via un compte d'organisation GitHub ou un groupe public sur gitlab.com ou une instance GitLab. Sur demande.
https://www.inria.fr/fr/la-production-logicielle	Quelques logiciels développés à Inria	Public, institutionnel	La liste exhaustive des codes Inria n'est accessible qu'en interne
http://www.cril.univ-artois.fr/softcats/lall/	Logiciels du laboratoire CRIL	Institutionnel	
https://www.labri.fr/valorisation/logiciels	Logiciels du laboratoire LaBRI	Institutionnel	
https://www.loria.fr/fr/productions/logiciels/	Logiciels du laboratoire LORIA	Institutionnel	
https://medialab.sciencespo.fr/outils/	Logiciels développés ou recommandés par le Medialab, laboratoire de Sciences-Po Paris	Institutionnel	
https://www.research-software.nl/	Projet du Netherlands eScience Center développé conjointement avec le Helmholtz Zentrum	Institutionnel	Référencement via l'inscription d'organisation
https://helmholtz.software/organisations/gfz?page=software	Sélection de logiciels développés par GFZ German Research Centre for Geosciences	Thématique	S'appuie sur l'outil de catalogue précédent
https://software.nasa.gov/	Catalogue des logiciels de la NASA	Institutionnel	
https://escape-ossr.gitlab.io/ossr-pages/	Infrastructure de référencement logiciels du projet européen ESCAPE	Institutionnel	S'appuie sur Zenodo
https://zbmath.org/software/	Catalogue de logiciels pour la communauté mathématique	Thématique	Un élément de Zentralblatt Math (ZbMath Open)
https://bio.tools/	Catalogue de logiciels pour la communauté bioinformatique et sciences de la vie	Thématique	Lié à l'infrastructure européenne Elixir
https://orms.mfo.de/	Catalogue pour la communauté mathématique	Thématique	En mode archive depuis 2021

https://www.osti.gov/doecode/biblio/68592	Logiciels développés par le Department of Energy étasunien (DOE)	Institutionnel	DOI, différents formats de citations
https://ascl.net/	Logiciels en lien avec l'astrophysique	Thématique	Workflow incluant une étape de modération
https://appdb.egi.eu/browse/software	Software market place pour l'infrastructure EGI	Public, institutionnel	Possibilité pour un utilisateur de signaler un logiciel
https://www.bioconductor.org/packages/release/BiocViews.html#___Software	Référencement de logiciels pour la communauté bioinformatique	Thématique	Essentiellement des paquets R en complément du CRAN
https://biii.eu/	Logiciels pour l'analyse d'images biologiques	Thématique	Base de données éditée sur le modèle wikipedia
https://www.nitrc.org/	Ressources pour la neuroimagerie	Thématique	Modération des ressources référencées
https://www.ontosoft.org/portal/#list	Plateforme de dépôt communautaire pour les logiciels en géosciences	Thématique	OntoSoft est un catalogue distribué où chaque site est maintenu par une communauté d'intérêt
https://sbgrid.org/software/	Logiciels pour la biologie structurelle	Thématique	Accès à un environnement complet d'exécution pour les membres du consortium
https://mloss.org/software/	Logiciels pour l'apprentissage automatique	Thématique	Associé au Journal of Machine Learning Research
https://nanohub.org/resources/tools	Plateforme dans le domaine des nanotechnologies, de la science des matériaux et des domaines connexes	Thématique	Catalogue très riche et interactif
https://www.comses.net/codebases/	Logiciels pour les sciences sociales et l'écologie (agent-based modeling)	Thématique	Peer review possible pour les auteurs le demandant