

L'ouverture des codes sources pour un enseignement supérieur et une recherche publique fiable

Une question du questionnaire visait à savoir si les personnes avaient connaissance de codes sources qui devraient être publiés, mais qui ne le sont pas encore. Même si peu de personnes donnent des exemples de logiciels non publiés (62,4 %), les personnes ayant répondu sont favorables à une ouverture du code source. Plusieurs raisons sont citées qui rejoignent des caractéristiques fortes constitutives de l'enseignement supérieur et de la recherche en tant que service public, mais également comme institution garante de productions de savoirs de qualité.

C.1 Le cadre légal applicable aux établissements de l'ESR

Les instituts de l'ESR sont soumis à un cadre légal favorable à l'open source :

- **en tant qu'administration** : tout logiciel, en tant que document produit ou reçu par un acteur de l'ESR dans le cadre de ses missions de service public de recherche, est soumis par principe au régime de l'open data. En effet, depuis la loi « Valter » de 2015, les centres de recherche ne peuvent plus appliquer leurs propres conditions de réutilisation des documents administratifs. Ils sont ainsi pleinement soumis au régime général de l'open data revu par la Loi pour une République numérique (LPRN) de 2016 qui a étendu le champ d'application de l'open data aux codes sources. Dès lors, la diffusion des codes sources devra se faire sous une licence open source et dans un standard ouvert, sans que des restrictions techniques ou juridiques puissent entraver son accès, sa modification, son analyse et sa redistribution. Au-delà de l'application du régime standard de l'« open data à la demande », les logiciels entrent dans la catégorie dite de l'« open data par principe » (ou par défaut) en tant que données « dont la publication présente un intérêt économique, social, sanitaire ou environnemental³³ » – ce qui impose un comportement proactif d'ouverture et de diffusion ;
- **en tant qu'acteur de l'ESR** : la loi n° 2013-660 du 22 juillet 2013 relative à l'enseignement supérieur et à la recherche dispose par ailleurs que les « logiciels libres sont utilisés en priorité » (codifié en l'article L. 123-4-1 du code de l'éducation) par le service public de l'enseignement supérieur dans la mise à disposition de « services et des ressources pédagogiques numériques ». Ces dispositions ont été prévues afin de favoriser une mutualisation dans le financement, l'achat et l'utilisation de logiciels au bénéfice des acteurs de l'enseignement supérieur et de la recherche³⁴.

³³ Article L 312-1-1 du CRPA.

³⁴ Un répondant au questionnaire a ainsi rédigé un commentaire en ce sens : « *ParcoursSup, tout l'algorithme d'affectation des élèves aux établissements, commentés et expliqués. Je pense qu'à partir du moment qu'un établissement public passe une commande de logiciels ceux-ci devraient être systématiquement mis sous licence Open Source (à part bien sûr cas vraiment spécifiques). Cela devrait être la condition primordiale.* »

Néanmoins, il semblerait que l'absence de sanction et de recours administratifs entraîne une faible application du cadre prévu par le législateur³⁵.

À noter qu'au niveau juridique, les licences open source CeCILL (CeCILL, CeCILL-B et CeCILL-C) ont été rédigées par le CNRS, le CEA et l'INRIA pour disposer de licences compatibles avec le droit français (face à un paysage open source majoritairement nord-américain) et, ainsi, sécuriser et accompagner les pratiques de l'open source au sein des centres de recherche. Ces licences ne sont aujourd'hui plus portées politiquement et humainement, rendant les chercheurs ou chargés de valorisation démunis devant ce choix, préjudicant aux objectifs initiaux. Un des chercheurs interrogés³⁶ notait qu'il avait utilisé les licences CeCILL mais avait arrêté, car « *cela posait problème avec l'international* ». D'autres pays, telle l'Allemagne, ont aussi historiquement produit leur propre licence, aujourd'hui abandonnée au profit des licences open source qui sont devenues des standards de fait. On peut noter aussi l'existence de la licence EUPL³⁷, qui ne figure pas néanmoins dans les licences autorisées au sein de l'administration française³⁸.

Recommandation 4 : Réaliser une étude pour évaluer la pertinence de l'emploi des licences françaises CeCILL et de la licence EUPL dans l'Enseignement Supérieur et de la Recherche et formuler des règles explicites quant à leur usage.

Outre ces enjeux de transparence, de « redevabilité » et de mutualisation, plusieurs commentaires soulignent également que l'ouverture des codes sources contribue à la qualité de la recherche produite, rejoignant les enjeux actuels portés sur la reproductibilité dans une démarche de science ouverte³⁹.

C.2 Science ouverte, reproductibilité et intégrité scientifique

L'ouverture des codes sources rejoint une problématique majeure au sein de la recherche qui concerne la qualité des savoirs produits et des modalités d'évaluation de la recherche tout autant que de son personnel (enseignant.e.s-chercheur.e.s, etc.). À ce propos, ont été cités également plusieurs fois en commentaires des entretiens et du questionnaire, l'enjeu de la reproductibilité en sciences nécessitant l'ouverture du « code scientifique » tout autant que des formations un système de reconnaissance et d'évaluation prenant en considération ces nouvelles pratiques.

35 Si ce n'est parfois quelques critères parfois favorisant les projets open source dans le cadre de certains investissements du MESRI (notamment dans la gestion de fonds du Programme d'Investissement d'Avenir PIA).

36 Entretien avec Konrad Hinsén, directeur de recherche au CNRS (Centre de Biophysique Moléculaire, Orléans).

37 cf. [https://www.iprhelpdesk.eu/sites/default/files/newsdocuments/European%20IPR%20Helpdesk%20Bulletin%20Issue%20\(26\).pdf#page=4](https://www.iprhelpdesk.eu/sites/default/files/newsdocuments/European%20IPR%20Helpdesk%20Bulletin%20Issue%20(26).pdf#page=4)

38 Décret n° 2017-638 du 27 avril 2017 relatif aux licences de réutilisation à titre gratuit des informations publiques et aux modalités de leur homologation <https://www.legifrance.gouv.fr/codes/id/LEGIARTI000034504991/2017-04-29/>

39 Jeffrey M. Perkel, « Challenge to Scientists : Does Your Ten-Year-Old Code Still Run ? », *Nature* 584, n° 7822 (24 août 2020) : 656-58, <https://doi.org/10.1038/d41586-020-02462-7>.

Points clefs questionnaire :

— Une idée récurrente dans les commentaires les codes sources des logiciels cités dans des articles devraient être publiés. L'enjeu semble être celui de la reproductibilité.

Ce point rejoint de façon plus globale les initiatives actuelles à l'échelle internationale, européenne et nationale de science ouverte (open access, plan de gestion des données, principes FAIR) soulignée notamment par la note d'opportunité du CoSo sur la valorisation des logiciels issus de la recherche. La publication du code source représente en ce sens une des briques du travail scientifique à partager afin de pouvoir tracer l'ensemble de la démarche scientifique :

- ces codes sources ici ne sont pas produits dans une optique de production logicielle ou de valorisation, c'est-à-dire dans le but de développer un outil utile à un grand nombre et qui a vocation d'être maintenu ;
- ces codes sources « *bouts de code* », « *scripts* », « *petits projets* », « *pipeline de données* », comme le rappellent certains commentaires et remarques de personnes interrogées, sont néanmoins la traduction en langage de programmation d'algorithmes nécessaires au traitement, à l'analyse et à la visualisation de données pour la production des résultats et des figures associées aux publications scientifiques⁴⁰.

Bien qu'associée à la notion juridique de logiciels⁴¹, cette distinction est importante à souligner car elle permet de mettre l'accent sur différents publics concernés par la production, diffusion et utilisation de ces éléments. En effet, ces « *scripts* » et « *codes scientifiques* » sont des noms usuels employés par les personnes les produisant (chercheurs, doctorants, etc.) dont les connaissances de développement « *logiciels* » sont différents d'ingénieurs dont le développement est le cœur de métier.

Des journaux scientifiques demandent désormais la publication du codesource (et également des jeux de données) qui ont servi à la production de résultats de recherche⁴². C'est le cas par exemple du journal IPOL⁴³ qui se présente comme un « *Open Science and Reproducible Research journal* ». L'objectif est de souligner l'importance des mathématiques comme source pour le design algorithmique et la reproductibilité en recherche. La revue ReScience⁴⁴, quant à elle encourage la réplication de travaux déjà publiés et promeut de « *nouvelles implémentations en open source* afin

40 Plusieurs citations peuvent être tirées du questionnaire : « Il y a des projets open source à long terme dans le but d'utiliser un projet utile. Et de l'autre côté il y a des bouts de codes pour un projet spécifique et ensuite on les oublie. On le rend public, mais il y a pas d'idées de maintenance et de services » et « Ce petit programme très vite obsolète et très conjoncturel, il a pas vocation d'avoir une longue vie. »

41 Pour un référencement des différentes définitions données aux logiciels de recherche (*research software*) comprenant ces « *scripts* »/codes scientifiques, voir : Teresa Gomez-Diaz et Tomas Recio, « On the Evaluation of Research Software : The CDUR Procedure », *F1000Research* 8 (26 novembre 2019) : 1353, <https://doi.org/10.12688/f1000research.19994.2>.

42 Extrait du questionnaire : « Diffuser le code source, c'est s'assurer que le travail peut être vérifié facilement. C'est une question de transparence, et il me semble que c'est même obligatoire dans certains domaines (au moins de fournir le code aux reviewers ».

43 <https://www.ipol.im/>

44 <http://rescience.org/x>

de s'assurer que les travaux originaux soient reproductibles ». ⁴⁵ En effet un de ses fondateurs en entretien ⁴⁶ note que les enjeux de reproductibilité nécessitent d'une part d'ouvrir les codes sources, mais également de s'assurer que l'environnement de travail disponible à ce moment puisse être retrouvé, et cela même 10 ans après. Cela implique ainsi des questionnements transverses sur le référencement et l'archivage et la constitution d'un patrimoine logiciel dont Software Heritage ⁴⁷ est une des initiatives clefs.

Recommandation 5 : Adopter une approche différenciée des bénéfices de l'open source en fonction des finalités de production des logiciels (codes scientifiques isolés ou production logicielle).

Recommandation 6 : Sensibiliser au partage des codes sources (et de jeux de données) étant associés aux résultats de recherche comme critères d'évaluation pour la publication dans une revue scientifique.

Pour l'ensemble des codes sources, des problématiques transversales se retrouvent telles : l'archivage, la maintenance, l'évaluation, mais aussi la valorisation possible des codes sources ⁴⁸.

	Production logicielle (outils et support)	Codes scientifiques/scripts
Métiers concernés	Ingénierie logicielle, ingénierie de recherche et d'étude	Métiers associés à la production de résultats scientifiques et évalués sur ces éléments (MCU, PU, DR, Post-doctorat et doctorat)
Finalité du code	Mutualisation et réutilisation du code, contribution et développement de nouvelles fonctionnalités, utilisation massive	Reproductibilité, preuve, traçabilité de la recherche, science ouverte
Enjeux	Valorisation, maintenance et soutenabilité, dynamique d'écosystème, visibilité	Acculturation aux pratiques open source documentation, citation, formation
Enjeux transversaux	Pérennisation, archivage, citation, référencement	
Ressources associées	Infrastructures de publication facilement interfaçables, référencement sur des plateformes, mises en place d'indicateurs, modèle de valorisation basée sur l'open (licences, contrat),	Personnes-ressources dans les laboratoires et en bibliothèque, infrastructures de publication interfacée à des archives, référencement avec identifiant pérenne,

Tableau visant à distinguer différentes finalités de l'ouverture du code source et des personnes concernées au sein de l'ESR

Cette distinction « manichéenne » entre production logicielle à maintenir et code source en soutien d'une démarche de science ouverte nécessite néanmoins d'être nuancée. D'une part, les codes sources servant à la production de résultats peuvent donner naissance lorsqu'ils sont stabilisés ou bien enrichis au développement de logiciel ou bien à des librairies supplémentaires pour des langages de programmation. D'autre part, ces deux éléments sont intimement liés pour la reproductibilité (le code source ne pouvant souvent qu'être reproduit si l'on retrouve le même environnement de travail (version du logiciel utilisé, dépendances, etc.).

⁴⁵ *ReScience/ten-years* (2019 ; repr., ReScience organization, 2020), <https://github.com/ReScience/ten-years> ; Konrad Hinsén et Nicolas P. Rougier, « ReScience », in *Open science, transparence et évaluation. Perspectives et enjeux pour les chercheurs* (Bordeaux, France : URFIST Bordeaux, 2017), <https://hal.archives-ouvertes.fr/hal-01573262>.

⁴⁶ Entretien avec Konrad Hinsén, directeur de recherche au CNRS (Centre de Biophysique Moléculaire, Orléans).

⁴⁷ <https://www.softwareheritage.org/?lang=fr>

⁴⁸ Pour plus d'informations, voir également la note d'opportunité du CoSO sur la valorisation des logiciels issus de la recherche décrit en détail les différents enjeux associés à la contribution, à la citation (métadonnées) à l'archivage.

Plusieurs commentaires du questionnaire et des entretiens mentionnent en ce sens que les enjeux de reproductibilité ou encore d'archivage sont des sujets et des pratiques nouvelles nécessitant un accompagnement. Un manque de méthodologie est souligné avec la nécessité de former les chercheurs à ces pratiques et aux enjeux de reproductibilité (environnement de travail, documentation du code, etc.). L'ouverture du « code scientifique » dans une démarche de science ouverte vise l'ensemble des communautés de chercheurs. Si les métiers d'ingénierie de recherche et d'étude et de développement logiciel sont de plus en plus familiarisés à l'open source et aux enjeux juridiques associés au sein de l'ESR, ce n'est pas le cas souvent des chercheur.e.s et étudiant.e.s (MCU, DR, PU, post-doctorat, doctorant) qui manquent de connaissances à ce sujet et de clefs de compréhension⁴⁹. À ce sujet, la procédure CDUR⁵⁰ a été proposée pour évaluer les logiciels de la recherche en quatre étapes : Citation, Dissémination (Dissemination), Utilisation (Use), Recherche (Research), pour analyser les conditions d'identification et de citation, les bonnes pratiques de diffusion et de (ré-)utilisation et l'évaluation de la recherche associée à ce type de logiciel.

Un accompagnement à ces pratiques (connaissances organisationnelles, juridiques, mais aussi génie logiciel (maintenance, suivi de version, test, etc.) est ainsi recherché. Le projet PLUME a ainsi proposé des formations et des ateliers pour sensibiliser à ces enjeux⁵¹. Un MOOC recherche reproductible a été également développé sur la plateforme FUN et soutenue par l'INRIA⁵². Un de ses créateurs⁵³ souligne néanmoins que les écoles doctorales ne sont souvent pas réceptives à ce nouveau genre de formation. Or l'open source joue également un rôle majeur dans la formation des nouvelles générations.

Recommandation 7 : Mettre à disposition des personnels de la recherche des formations, bonnes pratiques et outils en matière de publication de code source et les accompagner dans cette appropriation.

C.3 Open source et enseignement supérieur

De nombreux projets de recherche visent aussi à fournir les outils aux chercheur.e.s et ingénieur.e.s de demain. Outre l'utilisation de plus en plus fréquente de R et de Python dans les enseignements⁵⁴ d'informatique, de traitement et d'analyse des données (data science) (cf. B.3), plusieurs projets issus de travaux de recherche sont particulièrement connus⁵⁵ et utilisés dans le cadre de formation, que le projet provienne :

49 Une personne responsable de l'initiative Reproducible Research Oxford contactée par mail soulignait ainsi « *From what I have seen, it seemed to assume respondents would know that pieces of software they share on e.g. GitHub need a licence to allow reuse. From what we've seen (anecdotally), this may not be the case : people tend to think that pieces of software shared publicly are automatically Free and Open Source Software (FOSS) ».*

50 Gomez-Diaz et Recio, « On the Evaluation of Research Software ».

51 Les formations sont référencées sur le site du projet PLUME <https://www.projet-plume.org/patrimoine-logiciel-laboratoire>

52 Voir <https://www.fun-mooc.fr/courses/course-v1:inria+41016+self-paced/about>

53 Entretien avec Konrad Hinsén, directeur de recherche au CNRS (Centre de Biophysique Moléculaire, Orléans).

54 Des formations sont offertes au sein des universités et des Écoles mais aussi sous forme de MOOC accessibles en ligne.

55 Un certain nombre de logiciels employés au sein de l'ESR sont mentionnés sur le projet PLUME : [fiches_logiciels_dev_internes](#)

- des écoles (que ce soit à l'initiative des élèves tel VidéoLan, initialement développé par le bureau des élèves de l'École Centrale Paris, ou des instituts tels CHOCO, GPAC ou Open AIR développés par l'IMT) ;
- de centres de recherche, notamment du fait de l'activité d'enseignement réalisée par bon nombre de chercheurs (tels OrfeoToolBox⁵⁶ dans le domaine du géospatial, Gephi pour l'analyse de réseau, etc.).

L'implication dans la conception, le développement et l'usage de logiciels sur la base desquels sont formées les nouvelles générations (recherche, ingénierie) est déterminante pour participer au rayonnement de la recherche française. La diffusion en open source dans le cadre d'une gouvernance ouverte est une garantie facilitant l'implication et contribution de chercheurs étrangers. Ainsi la dimension communautaire internationale du logiciel open source GAMA Platform a-été décisive dans le choix de se former à cet outil par plusieurs équipes de recherche japonaises (Kyoto University et Ritsumeikan University).

On peut noter également dans le champ de l'enseignement à la programmation des projets déployés de façon indépendante en open source par des enseignants et chercheurs. C'est le cas de la plateforme hackinscience⁵⁷, un espace d'apprentissage de python avec la proposition de correction automatique d'exercices. Un des fondateurs⁵⁸ observe notamment son utilisation par des enseignants-chercheurs au sein d'Université aujourd'hui. Le projet OpenDreamKit⁵⁹ (projet européen H2020) gagne à être aussi souligné car il illustre l'importance de financement pour la création et le maintien d'environnement de recherche virtuel et collaboratif. Un des projets phares soutenus est Jupyter Notebook⁶⁰, un environnement de recherche jouant le rôle de « cahier de laboratoire », d'espace de documentation et de publication interactive de code pour des projets computationnels et de traitement de données. Ces environnements de travail apportent des briques opérationnelles afin de faciliter la reproductibilité de ce type de projet scientifique.

Plus globalement dans le champ de l'enseignement en ligne, le déploiement des MOOC a été majeur ces dernières années. En France, la plateforme FUN (France Université Numérique) développée à partir de 2013 repose sur la solution open EdX⁶¹ fondé par Harvard et le MIT en 2012. Celle-ci proposant selon un ancien développeur chez FUN⁶² « des avantages techniques et d'utilisation que Moodle⁶³ [projet open source d'apprentissage en ligne] négligeait un peu comme la problématique des interfaces utilisateurs qui sont souvent mal pensées dans le champ de l'open source ». Un travail conséquent a été nécessaire au début pour FUN afin d'adapter le projet pour « s'affranchir de la partie du code qui utilisait Amazon et YouTube. »⁶⁴. Ce même développeur notait des collaborations limitées avec la communauté EdX qui dénote d'un projet initial où le code est diffusé en open source mais sans être associé à une dynamique de contribution et communautaire comme cela peut-être le cas au sein de la communauté Moodle. À ce sujet, la

56 <https://www.orfeo-toolbox.org/>

57 <https://www.hackinscience.org/>

58 Entretien avec Julien Palard, co-fondateur de Hackinscience, développeur et formateur indépendant Python.

59 <https://opendreamkit.org/about/about/>

60 <https://jupyter.org/>

61 <https://open.edx.org/>

62 Entretien avec Laurent David, ancien développeur au sein du GIP FUN (2014-2017) et actuellement président de CALL Learning.

63 <https://moodle.org/?lang=fr>

64 Le code source de FUN est disponible sur Github : <https://github.com/openfun>

problématique du développement et du maintien de projets logiciels comme support de l'ESR a été adressée et implique de considérer l'open source comme élément clef des infrastructures de l'enseignement supérieur et de la recherche.

C.4 Les infrastructures support de l'ESR

Plusieurs commentaires insistent sur le fait que la production logicielle est issue de l'investissement d'argent public et devrait ainsi être ouverte. Cela a été mentionné pour le cas de la plateforme d'archivage HAL en commentaire du questionnaire⁶⁵. Un autre exemple cité était l'ouverture des algorithmes entiers de ParcoursSup en précisant l'importance d'une documentation associée pour expliciter ces algorithmes⁶⁶. Plusieurs réponses soulignent également la nécessité de logiciels libres pour les infrastructures de recherche et d'enseignement supérieur. Cela concerne aussi bien des logiciels employés au sein des laboratoires ou de l'enseignement, qu'ils soient généralistes (à la place des suites Microsoft) ou bien plus spécifiques.

Le logiciel, ainsi plus qu'un objet de recherche, un artefact ou un outil, est aussi à la base même des infrastructures de soutien au fonctionnement de l'ESR, qui représenterait une quatrième fonction, ou catégorie, majeure à considérer. En anglais, le terme d'« *Open Scholarly infrastructures* » ou d'« *Open Source Publishing Tools and Platforms* » est employé pour mentionner les infrastructures soutien à la publication, à l'archivage des produits de la recherche (données, publications, etc.)^{67 68}. Cette prise en considération est devenue majeure depuis quelques années avec la thématique plus générale des « *Digital Infrastructures* » et de leur durabilité (*sustainability*) qui bénéficie de financement conséquent notamment aux États-Unis⁶⁹. Outre le GIP (Groupement d'intérêt public) FUN en open source, le GIP AMUE (Agence de Mutualisation des Universités et Établissements) dédié à la mutualisation des outils support des domaines d'activités de l'ESR se positionne d'ores et déjà dans une approche de co-construction et de mutualisation avec les universités pour développer des projets open source⁷⁰. L'open source étant un levier pour favoriser une mutualisation des

65 Le CCSD a récemment publié le code de HAL, désormais disponible sur [github](https://github.com). De nouveaux usages au sein de l'administration sont en cours d'exploration. La gouvernance de HAL est actuellement en cours de reconfiguration pour développer une gouvernance partagée et un modèle de financement pérenne : <https://www.enseignementsup-recherche.gouv.fr/cid154220/hal-une-gouvernance-partagee-et-un-modele-de-financement-perenne.html>

66 Avant ParcoursSup, la publication du code de APB (Admission Post-Bac) avait elle-même fait très largement réagir, la transparence offerte par l'open source rendant en effet visible des prises de décisions automatisées par l'outil (aujourd'hui remplacées par des interventions humaines). La documentation et le code sont ouverts sur framagit : <https://framagit.org/parcoursup/algorithmes-de-parcoursup>

67 John W Maxwell et al., *Mind the Gap : A Landscape Analysis of Open Source Publishing Tools and Platforms*, 1^{re} éd. (PubPub, 2019), <https://doi.org/10.21428/6bc8b38c.2e2f6c3f>.

68 Neylon Cameron, « Principles for Open Scholarly Infrastructures », 2015, <https://cameronneylon.net/blog/principles-for-open-scholarly-infrastructures/>.

69 La Ford Foundation par exemple propose des financements de projets portant que les infrastructures numériques. Le livre de Nadia Eghbal, *Roads and Bridges : The Unseen Labor Behind Our Digital Infrastructure* (Ford Foundation, 2016).

70 Un groupe de travail de veille et de prospective a ainsi été constitué pour réfléchir sur les choix de technologies à mettre en œuvre compatible open source tout en questionnant le modèle économique associé. Ce point nécessite une transition économique pour ce Groupement d'Intérêt Public afin de s'adapter, mais également repenser le rapport avec les établissements adhérents et autres acteurs numériques de l'ESR selon David Rongeat (responsable numérique AMUE) et Bertrand Mocquet (expert numérique AMUE). Voir la [fiche « co-construction et mutualisation »](#) de l'AMUE.

initiatives et proposer une gouvernance numérique de l'ESR basée sur ces principes. Plus encore, l'ensemble des logiciels administratifs (gestion, comptabilité, édition) et leur « composante métier » pourraient bénéficier d'une telle mutualisation.

Recommandation 8 : Mutualiser les moyens logiciels mis en œuvre dans le développement et la maintenance des infrastructures nécessaires à la publication des codes sources de l'ESR.