



# *Article vs. Logiciel : questions juridiques et de politique scientifique dans la production de logiciels*

Teresa Gomez-Diaz

Laboratoire d'informatique Gaspard-Monge – PLUME

18 octobre 2011

<http://www.projet-plume.org/ressource/article-vs-logiciel>

Document distribué sous licence CC by-nc-nd : <http://creativecommons.org/licenses/by-nc-nd/2.0/fr/>

## **Introduction**

Ce document continue et complète les études déjà publiées sur PLUME (voir le thème PLUME : [Patrimoine logiciel d'un laboratoire](#)) sur les logiciels développés dans les laboratoires et autres entités de la communauté d'enseignement supérieur et recherche. Il répète des choses déjà incluses dans d'autres documents PLUME ; elles sont reprises ici parce qu'on souhaite avoir une vision la plus complète possible, mais aussi parce qu'on change la perspective de l'étude.

L'objectif de ce document est donc l'étude de cet objet complexe, "logiciel d'un laboratoire" (ou d'un organisme d'enseignement ou de recherche) pour comprendre les conditions de développement dans les laboratoires, et les questions que ces développements soulèvent. Les questions étudiées sont d'ordre juridique ou d'ordre décisionnel.

D'après le travail mené dans PLUME depuis 2007, nous pouvons classer les différents points à prendre en compte lors du développement et de la diffusion d'un logiciel dans notre communauté comme suit :

- formation, support
- techniques : outils, briques, méthodes de développement, fiabilité, ...
- communauté : gestion, prise de décisions, communication interne, résolution de conflits, communication externe, ...
- (v) administratifs : dépôt APP, ...
- (v) financement : contrats, postes, projets, ...
- (v) juridiques : propriété intellectuelle et droit d'auteur, licences, brevets, contrats, ...
- (v) politique scientifique : des laboratoires, des tutelles, propriété du code, contrats quadriennaux, reproductibilité de la recherche, libre accès, définition de procédures (de recensement, de diffusion, d'évaluation, de gestion du patrimoine), ...
- politique/législatif : définition de standards, législation sur les brevets, ...

Les points signalés avec une (v) sont liés à la valorisation des logiciels et de la recherche. Dans la plupart de ces éléments on peut mentionner les **bonnes pratiques** qui sont à connaître, à utiliser, ou tout simplement à mettre en place. Ces problèmes se posent indépendamment du type de licence qu'on choisira pour la diffusion du logiciel ou du modèle d'exploitation.

Ce document considère l'étude de deux aspects : le point de vue juridique, lié au droit d'auteur et aux licences, et le point de vue de politique scientifique. Pour cela, **je compare les logiciels aux publications de recherche, plus particulièrement aux articles de recherche, pour mieux comprendre tous les aspects concernant la propriété intellectuelle, mais aussi pour saisir les aspects orientés vers la valorisation, l'évaluation de la recherche et la définition des politiques scientifiques.** Lorsqu'on étudie les aspects juridiques, ce document fait référence au [Code de la propriété intellectuelle \(CPI\) en France](#).

Je ne suis pas juriste, mais je m'intéresse à ce sujet depuis qu'une "mission logiciels" relative à la visibilité des logiciels du laboratoire m'a été confiée au sein de mon unité, le laboratoire d'informatique Gaspard-Monge (LIGM). Cette expérience s'est renforcée depuis mon intégration dans PLUME en décembre 2008. D'autres documents PLUME liés à cette étude sont classés dans le thème PLUME [Patrimoine logiciel d'un laboratoire](#).

Des versions de préparation de ce document ont été relues par Pascal Janots, responsable du [SAIC de l'UPEMLV](#) (service de valorisation de la recherche de l'université Paris-Est Marne la Vallée), [Geneviève Romier](#) (responsable thématique PLUME), [Laurette Chardon](#) (GREYC), [Eric Barbry](#) (Cabinet Alain Bensoussan), [Konrad Hinsén](#) (CBM), [Florence Petit](#) (IGM), [Véronique Baudin](#) (responsable thématique PLUME), [Jean-Luc Archimbaud](#) (rédacteur en chef PLUME). Merci à ces personnes (et d'autres qui resteront anonymes), elles m'ont aidée à valider ce texte tant sur le fond que sur la forme.

Le contenu de ce document a fait l'objet de plusieurs présentations tout au long des années 2010 et 2011, principalement lors de FOSSA 2010 (Grenoble) et Solutions Linux/Open source 2011 (Paris). Il donne parfois mon opinion, mais il tente surtout de répondre aux nombreuses questions qui se posent sur ce sujet actuellement. Aucune erreur dans ce texte ne peut être attribuée aux relecteurs.

Si vous souhaitez réagir au contenu de ce document, ou y ajouter des nouveaux points, vous pouvez écrire des commentaires (en bas de cette page) ou contacter l'auteur.

### **Pour les lecteurs pressés**

Ce document se résume succinctement dans les trois tableaux suivants :

- [les 19 points de comparaison entre articles scientifiques et logiciels](#),
- [les points « aspects légaux »](#),
- [les points « aspects de politique scientifique »](#).

Vous pouvez y repérer les points qui vous intéressent puis vous y rapporter dans le texte, chaque point est traité de façon assez indépendante du reste, mais il faut lire d'abord le point **Définition** : il est traité en premier et établit le cadre de comparaison.

## **Les points de comparaison entre articles scientifiques et logiciels**

Ce document étudie 19 points de comparaison entre les articles scientifiques et les logiciels. Ces points sont classés en deux grandes sections : les aspects juridiques et les aspects décisionnels et relatifs à la politique scientifique. Ces 19 points sont présentés de manière synthétique dans le tableau suivant.

Article vs. Logiciel	
Aspects légaux	Aspects relatifs à la politique scientifique
<u>Droit d'auteur</u> <u>Oeuvre</u> <u>Auteurs</u> <u>Propriétaires</u> <u>Dates</u> <u>Évolution de l'oeuvre</u> <u>Travaux précédents</u> <u>Diffusion</u> <u>Droits</u> <u>Licences</u>	<u>Définition</u> <u>Signature</u> <u>Références</u> <u>Liste des oeuvres d'un laboratoire</u> <u>Libre accès</u> <u>Validation</u> <u>Qualité et évaluation</u> <u>Motivation</u> <u>Objet</u>

Table 1. Les 19 points de comparaison entre articles et logiciels.

Les points sont ordonnés selon une certaine logique. Dans les aspects juridiques, les points relatifs au droit d'auteur sont traités avant ceux associés aux licences, le point Auteurs est à traiter avant le point Propriétaires, les points Diffusion, Droits et Licences sont à étudier ensemble. L'ordre n'est pas trop important dans la section de politique scientifique : le point **Définition** est le plus important de cette liste, il est aussi utilisé pour donner le point de départ de ce document.

## Le cadre de comparaison

Le point **Définition** est classé dans les aspects de politique scientifique, il est traité en premier pour fixer le cadre de la comparaison et définir les objets à comparer.

### Définition

En général, un article de laboratoire est un article publié dans une revue scientifique dont un des signataires appartient à un laboratoire. C'est, en général, un objet bien compris qui ne pose pas de question.

Un "logiciel de laboratoire" continue à être un objet mal compris : j'entends par logiciel de laboratoire tout programme ou fragment de programme utile pour faire avancer la recherche et qui a été produit avec la participation d'un ou plusieurs membres du laboratoire. Ce qui est utile pour faire avancer la recherche comprend un large spectre. Cette définition peut s'élargir pour inclure des programmes développés pour des besoins d'enseignement, pour améliorer la gestion (contrats, bibliographie, recrutement, ...).

Pour mieux faire le parallèle entre article et logiciel, on considère dans ce document uniquement les programmes qui ont été développés pour étudier un objet scientifique ou faire un calcul et qui ont donné des publications scientifiques (donc avec un "article de laboratoire" associé). On utilisera à ce propos le terme "logiciel de recherche" (comme traduction du terme anglais "scientific software") et le terme "logiciel académique" pour désigner un logiciel produit au laboratoire dans son interprétation la plus large.

Un logiciel de recherche reste, malgré la définition donnée, un objet mal défini : les quelques lignes écrites pour vérifier un calcul dans des logiciels comme Matlab, Sage ou autres ne sont pas facilement comparables avec un logiciel réalisé par plusieurs personnes pendant des années pour comprendre et accompagner le développement d'une théorie. Des difficultés similaires apparaissent

lors de la comparaison des publications scientifiques. De façon analogue au traitement d'articles dans un laboratoire, les responsables des équipes et des laboratoires décident quand un programme (ou un article) fait partie de la production scientifique du laboratoire.

Mais cette décision peut ne pas être réduite au périmètre du laboratoire puisque, en général, les tutelles d'un laboratoire sont les propriétaires des logiciels qui y sont produits (voir le point [Propriétaires](#)) ; la définition doit donc être établie en fonction de la politique scientifique du laboratoire et doit prendre en compte l'avis et la politique scientifique des tutelles.

## Aspects légaux

On étudie dans cette section des aspects liés au droit d'auteur et aux licences, en regardant les différences et similitudes entre les articles et les logiciels. On introduit aussi du vocabulaire juridique. Pour approfondir ces notions légales il faut se référer au Code de la propriété intellectuelle (CPI) ou aux spécialistes.

### Droit d'auteur

Les droits protégés par le Code de la propriété intellectuelle (CPI) sont automatiquement associés à l'auteur lors de la création de l'oeuvre, sous condition de son originalité. L'oeuvre doit aussi être mise en forme (texte, video, tableau, ...) : les idées, les concepts ne peuvent pas être protégés.

Le droit d'auteur ne s'applique pas de la même façon aux articles et aux logiciels, ces derniers bénéficient d'un traitement spécial.

Pour les articles comme pour les oeuvres en général, les droits d'auteur se divisent en deux classes :

- Droits moraux : ces droits concernent la maîtrise de l'oeuvre ; ce sont des droits imprescriptibles, inaliénables, incessibles, et en général ils sont associés à des personnes physiques (les auteurs ou ses héritiers). Il y en a quatre :
  - Droit à la paternité, relatif à la mention de l'auteur.
  - Droit de divulgation, relatif au moment et aux conditions de livraison.
  - Droit de repentir et de retrait, qui permet de revenir sur la cession des droits (patrimoniaux) d'une oeuvre.
  - Droit au respect de l'oeuvre, qui permet de s'opposer aux modifications.
- Droits patrimoniaux : ces droits concernent l'exploitation de l'oeuvre, et sont des droits monnayables, cessibles, temporaires. On considère qu'il y a deux types d'exploitation : la représentation (par exemple d'une oeuvre de théâtre) et la reproduction (musique sur CD par exemple). Ces droits sont souvent associés à des personnes morales (à la suite des cessions effectuées par les auteurs) qui peuvent avoir (en fonction des contrats négociés) un monopole d'exploitation de l'oeuvre. On parle alors de détenteurs des droits patrimoniaux.

Pour les logiciels, il y a également des droits patrimoniaux et des droits moraux mais avec des différences par rapport à l'application du droit aux articles et aux oeuvres en général. Les différences que je souhaite mentionner dans ce document sont les suivantes :

- Droits moraux (article L. 121-7 du CPI) : l'auteur ne peut (sauf stipulations contraires) s'opposer à la modification de l'oeuvre ou exercer son droit de repentir ou de retrait, sauf si ces modifications causent un préjudice à l'honneur ou à la réputation de l'auteur.
- Droits patrimoniaux (article L. 113-9 du CPI) : ces droits sont (sauf stipulations contraires) dévolus à l'employeur.

Les droits moraux ne fonctionnent pas de la même façon pour les logiciels que pour les oeuvres en général. On parle donc de droits d'auteur (moraux) réduits pour les logiciels :

- le droit à la paternité reste intact ;
- le droit de divulgation remonte automatiquement à l'employeur (dans le cas des auteurs salariés, voir le point [Propriétaires](#)) ;
- le droit de repentir devient non applicable ;
- le droit au respect de l'oeuvre est réduit aux cas des modifications préjudiciables à l'honneur ou à la réputation de l'auteur.

Lorsqu'on ne connaît plus de personne physique (auteur ou héritier d'auteur) associée aux droits moraux d'une oeuvre, on parle d'oeuvre orpheline. Les oeuvres (article L. 123-1 du CPI) entrent dans le domaine public lorsque la durée des droits patrimoniaux est atteinte, c'est-à-dire 70 ans après la mort de l'auteur d'un article. Le guide pratique [PME : pensez propriété intellectuelle !](#) indique que la durée des droits patrimoniaux pour les logiciels est de 50 ans, donc par exemple, pour qu'un logiciel passe dans le domaine public en 2011, il faut que son auteur (s'il y a un unique auteur) soit décédé en 1961. Après ce délai l'oeuvre devient librement exploitable (en ce qui concerne les droits patrimoniaux), il n'y a plus de monopole d'exploitation. Un logiciel entré dans le domaine public n'est pas nécessairement un logiciel libre.

Il faut aussi remarquer que ce terme "logiciel du domaine public" est parfois utilisé, à tort, pour des logiciels avec [des licences du type BSD ou sans copyleft](#). Rappelons aussi qu'en droit anglo-saxon, il y a beaucoup plus de droits qui sont cessibles qu'en droit français, où toute cession comprenant des droits moraux est contraire au droit d'auteur. Le terme "logiciel du domaine public" a donc tout son sens en terminologie anglo-saxonne. Le terme [res derelictae](#) semble aussi approprié.

## Oeuvre

Dans le cas des articles, l'objet protégé par la loi est évident : c'est l'article.

Lorsqu'on parle d'un logiciel il faut savoir que le périmètre de protection est plus large et comprend le programme (code source, code compilé), le matériel de conception, la documentation, les interfaces, son titre.

Le matériel de conception et le titre d'un article sont aussi protégés par le droit d'auteur, mais à ma connaissance, ce type de question pose rarement de problème dans la communauté scientifique.

## Auteurs

D'un point de vue légal, l'auteur d'une oeuvre est celui qui la réalise : celui qui l'écrit pour une oeuvre écrite, celui qui peint pour une peinture, ...

Les auteurs d'un article sont les signataires du document publié : l'article L. 113-1 du CPI indique que la qualité d'auteur appartient, sauf preuve contraire, à celui (ou à ceux) sous le nom de qui l'oeuvre est divulguée. Ces personnes ont, d'un point de vue légal, le même pourcentage de participation à l'oeuvre, même si parfois l'ordre de la liste d'auteurs est interprété autrement dans la communauté scientifique.

En ce qui concerne le logiciel, déterminer si un contributeur à un logiciel en est l'auteur peut être un problème légal (voir par exemple [Méthode pour tracer la propriété intellectuelle dans des codes logiciels](#)). Ce n'est pas le cas d'une grande partie des logiciels des laboratoires, mais il peut y avoir des cas compliqués, en voici quelques-uns.

- Des contributeurs qui n'ont corrigé que quelques lignes de code.
- Des contributeurs qui ont beaucoup apporté à un logiciel mais leur partie a été entièrement réécrite par une autre personne quelques années plus tard.
- Des développeurs dont la contribution a été filtrée par un responsable du logiciel : cette contribution peut être plus ou moins modifiée lors de son intégration au code principal.

- Des personnes travaillant en équipe qui expliquent et donnent les instructions pour écrire un programme : sans leur apport le programme n'existerait pas.
- Les codes générés automatiquement : qui en est l'auteur ? Est-ce le programme qui les génère, l'ordinateur où ils ont été générés, l'auteur du code générateur ? Si la génération automatique du code peut se comprendre (d'un point de vue légal) de façon similaire à la compilation d'un code, l'auteur du code générateur est aussi auteur du code généré automatiquement. Je souhaite citer ici le cas de l'ordinateur [Shalosh B. Ekhad](#) qui [signe des articles publiés](#).
- Les codes traduits (par exemple de Fortran à C) : l'auteur du code original, est-il l'auteur du code traduit ? La réponse est affirmative dans ce cas, mais compliquée en fonction de nouveaux apports qui peuvent être effectués lors de la traduction.

On pourra trouver bien d'autres exemples pour illustrer la nature intrinsèquement complexe de la définition d'auteur d'un logiciel. De plus, la génération de droits d'auteur pour ces contributions est soumise, comme pour toute oeuvre, à la condition d'originalité.

Pour éviter des possibles problèmes légaux, il convient d'avoir un document, actualisé régulièrement, qui mentionne les auteurs du logiciel et leur pourcentage de participation. Avoir un document daté et signé est un plus légal.

Pour tester la compréhension de ce concept difficile, on a posé la question suivante : est-il possible de faire un logiciel de façon anonyme ? L'anonymat ou l'utilisation d'un pseudonyme sont liés au droit moral de paternité. Cela correspond donc à un choix exclusif de l'auteur.

Sur cette question d'anonymat je souhaite citer les oeuvres publiées du célèbre mathématicien [Nicolas Bourbaki](#) : nom de plume d'un groupe de mathématiciens qui ont travaillé de façon anonyme et dont les oeuvres ont révolutionné la pensée mathématique mondiale. Une telle liberté scientifique, est-elle possible lorsqu'on parle d'un logiciel ? Sera-t-elle nécessaire un jour ?

Un autre exemple qui teste les limites du droit d'auteur est la célèbre encyclopédie [Wikipédia](#).

## Propriétaires

Pour un article, les auteurs sont propriétaires ou détenteurs des droits patrimoniaux de l'oeuvre d'un point de vue légal et partagent le même pourcentage de propriété. Par exemple, ce sont les auteurs qui signent les contrats de transfert des droits (d'exploitation) vers les éditeurs.

Cela ne s'applique pas aux logiciels, qui font l'objet d'un traitement spécial par le droit d'auteur. Les propriétaires d'un logiciel sont ses détenteurs des droits patrimoniaux. L'article L. 113-9 du CPI stipule : "Sauf dispositions statutaires ou stipulations contraires, les droits patrimoniaux sur les logiciels et leur documentation créés par un ou plusieurs employés dans l'exercice de leurs fonctions ou d'après les instructions de leur employeur sont dévolus à l'employeur qui est seul habilité à les exercer."

Dans notre communauté, cela indique en général que les tutelles d'un laboratoire sont les propriétaires des logiciels, et ce sont elles qui doivent prendre les décisions qui impliquent les droits patrimoniaux associés aux logiciels : par exemple, le choix de la licence d'un logiciel, son exploitation, sa diffusion, ... Lorsque plusieurs auteurs d'établissements différents participent à un même logiciel, le pourcentage de propriété d'une tutelle se dérive du pourcentage de participation des auteurs employés par cette tutelle.

Par ailleurs, les auteurs qui n'ont pas un régime salarié (stagiaires, professeurs émérites, ...) sont aussi propriétaires du logiciel, en fonction de leur participation, sauf s'il existe des dispositions qui précisent ces droits dans un contrat ou autre document, (voir [Méthode pour diffuser un logiciel de laboratoire : recommandations juridiques et administratives](#)).

Les contrats (de collaboration, de commande, ...) qui sont parfois établis peuvent indiquer qui est le propriétaire du logiciel obtenu.

En conclusion, les détenteurs des droits patrimoniaux ou propriétaires d'un logiciel sont établis en fonction de :

- les auteurs,
- leur statut et leur mode de participation,
- les contrats entre employeurs et les salariés qui participent au développement,
- tout autre contrat qui traite du développement de logiciel (par exemple les contrats de collaboration, de commande),
- dans le cas particulier d'un laboratoire, les conventions entre tutelles (ou le droit conférant à un mandataire unique, voir l'article R. 611-13 du CPI) peuvent décider du traitement de ces droits patrimoniaux.

## **Dates**

Pour un article, les dates importantes sont la date de soumission et la date de publication.

Pour un logiciel, une des dates importantes est celle associée au matériel de conception. En effet une oeuvre est protégée par le droit d'auteur si elle est originale ; déterminer si une oeuvre est originale peut demander l'intervention de juges et autres experts. D'autres dates importantes pour un logiciel sont celles des versions qui indiquent des fonctionnalités importantes ajoutées ou des évolutions notables, toujours soumises par le droit d'auteur à une condition d'originalité.

Le matériel préparatoire peut servir à dater un article, mais en général la communauté scientifique évite cette question en publiant rapidement une première version du travail.

Au milieu du bazar que peut constituer le développement d'un logiciel (voir l'article [La Cathédrale et le Bazar](#)), on peut comprendre les difficultés auxquelles les juges font face lorsqu'ils doivent prendre des décisions : l'existence de dates clairement établies et d'une valeur légale irréfutable est importante. Ceci explique pourquoi les dépôts APP, les registres IDDN et autres (voir [L'agence de protection des programmes \(APP\) et le registre Inter Deposit Digital Number \(IDDN\)](#)) sont appréciés par les tutelles et le monde juridique. D'autres moyens existent, mais il faut veiller à leur valeur juridique.

## **Évolution de l'oeuvre**

Sans aucun doute l'évolution d'un article est en général un autre article, c'est-à-dire une nouvelle oeuvre, avec ses auteurs, son titre et son contenu, tout-à-fait indépendante de l'article précédent (d'un point de vue juridique).

L'évolution d'un logiciel est en général une nouvelle version du même logiciel. S'agit-il d'une nouvelle oeuvre (indépendante de la précédente, dans le même sens que pour les articles) ? Je n'ai pas de réponse à cette question, et je crois pouvoir affirmer que dans le monde du logiciel, beaucoup d'aspects restent encore mal compris.

Une chose est claire, à chaque nouvelle version d'un logiciel, il faut tout revoir : la liste d'auteurs et leur pourcentage de participation, la description de l'oeuvre et l'originalité des nouveaux apports, les propriétaires et les dates associées.

## **Travaux précédents**

Dans un article on cite les oeuvres précédentes et les oeuvres liées au travail en cours. Cela ne pose pas en général de problème juridique.

Un logiciel peut incorporer et modifier d'autres briques logicielles. Les problèmes légaux peuvent apparaître lorsque l'on étudie la compatibilité des licences des briques utilisées et l'héritage de licence envers le logiciel en cours. Les licences et droits d'utilisation doivent être regardés avant l'utilisation des briques, puisque le fait de les utiliser implique que leurs licences ont été acceptées et que le contrat que ces licences représentent est en application.

Des logiciels comme [FOSSology](#), [OSLC](#) et les formats [SPDX](#) et [Open source cartouche](#) peuvent être utiles pour éclaircir les questions de compatibilité et héritage de licences dans la gestion des cas compliqués.

## **Diffusion**

La diffusion d'un article se fait normalement par les éditeurs d'une revue. Depuis quelques années, la diffusion se fait aussi en utilisant le web, par les éditeurs, les auteurs ou bien par le moyen des pages web personnelles et des sites comme [ArXiv](#), [HAL](#) et autres.

Les logiciels se diffusent souvent par le web, en utilisant parfois des [forges](#). Avant sa diffusion, il faut réfléchir à [la licence](#) qui va accompagner le code.

D'un point de vue légal, diffusion signifie donner l'oeuvre à quelqu'un, peu importe le support (citons par exemple l'affaire sur la FreeBox en cours de jugement, voir par exemple [Séminaire "Construire son projet sur du libre : quelles précautions prendre ?"](#)). Il est souhaitable d'avoir des licences en place ou bien des contrats de collaboration avant tout échange de logiciel.

## **Droits**

Tout le monde peut lire un article. Il peut y avoir une barrière scientifique, mais il n'y a pas de barrière légale. Copier une oeuvre peut être illégal, à la différence de la citer (article L. 122-5 du CPI).

De façon similaire, il est possible de lire un code, tout au moins essayer. Il peut y avoir aussi des barrières scientifiques et peut être de langage (de programmation). Mais l'utilisation d'un code sans avoir le droit explicitement donné (voir l'article L. 335-2 du CPI) relève de la contrefaçon, et similairement pour la modification d'un code, sa copie, sa redistribution, ... Il est utile de donner des licences aux logiciels pour établir clairement les droits octroyés.

## **Licences**

On utilise de plus en plus de licences du type [CC](#) et autres pour les documents diffusés par le web, même pour des articles publiés dans des revues scientifiques, voir par exemple la politique de copyright de la revue [Logical Methods in Computer Science](#) ou la politique [Oxford Open](#).

[On donne des licences aux logiciels, en particulier des licences libres](#). Toute licence qui n'est pas libre est une licence propriétaire. Il existe donc des logiciels dits libres ou propriétaires en fonction de la licence qui les accompagne. Un logiciel peut être accompagné de plusieurs licences et donc, il peut être libre et propriétaire à la fois. L'utilisation de licences multiples donne aux utilisateurs le choix de licence à utiliser et aide à simplifier des problèmes de compatibilité.

De plus, différentes parties d'un logiciel peuvent être associées à des licences différentes (comme les interfaces graphiques). Le terme "logiciel semi-libre" est parfois utilisé dans ce cas, je préfère l'éviter : mal défini, il reste ambigu et peut donner lieu à des interprétations contradictoires.

Les licences font intervenir des droits patrimoniaux, elles doivent se mettre en place avec l'accord de tous les détenteurs de ces droits (qui demandent souvent, en pratique, l'avis des auteurs).

Les licences donnent des droits et imposent des obligations qui sont à respecter.



Nées dans le milieu scientifique, les licences libres sont particulièrement utiles pour les logiciels de recherche (voir par exemple ["Pourquoi diffuser un logiciel développé dans un laboratoire ou une université avec une licence libre ?"](#)). Prenons par exemple le cas où des stagiaires sont intervenus dans un développement, ils sont détenteurs des droits patrimoniaux au même titre que les employeurs des collaborateurs salariés. Dans ce cas, la licence libre permettra l'évolution du logiciel au laboratoire et aussi à l'extérieur (si les stagiaires souhaitent continuer leur projet après la fin du stage) dans le respect des droits de tous les ayants droits impliqués dans ce logiciel. Un autre cadre où les licences libres sont particulièrement utiles est celui des collaborations internationales ou encore celui où les développeurs prévoient une forte mobilité (entre employeurs).

Certains juristes semblent ne pas décider si les licences libres sont ou non des contrats. D'autres juristes (voir par exemple la thèse "Les oeuvres libres" de Mélanie Clément-Fontaine à l'Université de Montpellier 1, 2006) affirment que ce sont des contrats entre les auteurs et les utilisateurs ou les lecteurs : ce sont des contrats passés à titre gratuit et qui ont une vocation internationale. Des jurisprudences en France confirment que c'est de cette deuxième façon que les juges comprennent et appliquent ces licences (voir par exemple ["Les logiciels libres : soumis au droit d'auteur, dans un contexte international, une jurisprudence en émergence, des défis à relever"](#) RMLL'09).

### Tableau récapitulatif sur les aspects légaux

Voici un tableau qui récapitule les 10 points traités :

Article vs. Logiciel : aspects légaux		
	Article	Logiciel
<b>Droit d'auteur</b>	oeuvre protégée : droits moraux, droits patrimoniaux	oeuvre protégée mais <b>traitement spécial CPI</b> : droits moraux <b>réduits</b> , droits patrimoniaux <b>dévolus à l'employeur</b>
<b>Oeuvre</b>	article	code source, code objet, documentation, matériel de conception, ...
<b>Auteurs</b>	signataires, même %	notion complexe, <b>problème légal</b> , établir un <b>% de participation</b>
<b>Propriétaires - Droits patrimoniaux</b>	auteurs, même %	tutelles en général, mais dépend du <b>régime salarié</b> des auteurs et des <b>contrats</b> (de collaboration, de commande, ...)
<b>Dates</b>	soumission, publication	matériel de conception, <b>versions</b>
<b>Évolution de l'oeuvre</b>	oeuvre indépendante	oeuvre indépendante ? il faut <b>revoir</b> auteurs, dates, licences, ...
<b>Travaux précédents</b>	références, citations	briques logicielles : <b>compatibilité</b> et <b>héritage de licences</b>
<b>Diffusion</b>	éditeur, web	web, forges <b>besoin de licence</b>
<b>Droits</b>	lire, citer, ne pas copier	lire, ne pas utiliser, ne pas modifier, ... <b>besoin de licence</b>
<b>Licences</b>	Creative Commons (web)	droits et obligations, libres ou propriétaires

Table 2. Les 10 points « aspects légaux » de comparaison entre articles et logiciels.

Ce tableau nous aide à mieux comprendre les différences entre le droit d'auteur des articles et le droit d'auteur des logiciels. On observe que le modèle légal des articles fonctionne bien dans les laboratoires et ne pose pas de problème en général.

Le cadre légal des logiciels est un peu plus compliqué, les logiciels bénéficient d'un traitement spécial par le droit d'auteur. La notion d'auteur d'un logiciel peut être complexe et donc relever d'un problème légal. En fonction du nombre d'auteurs et du cadre de collaboration, la gestion des droits patrimoniaux peut devenir aussi complexe. D'autres problèmes juridiques seront liés à la contrefaçon.

Un des moments les plus importants de la vie d'un logiciel est celui de la diffusion. Il faut réfléchir aux licences qui vont être données au code. Mais avant de s'occuper des licences, il faut avoir mis au clair tous les aspects liés au droit d'auteur (voir par exemple [Méthode pour diffuser un logiciel de laboratoire : recommandations juridiques et administratives](#)). Les licences doivent être mises en place avant la diffusion de l'oeuvre. C'est une question d'actualité, tant pour les articles (voir par exemple [Journées du réseau Médecin, Grenoble, MSH Alpes, 5 et 6 avril 2011 - Mutualiser les bonnes pratiques : quelles réalités et quel avenir pour l'édition scientifique publique ?](#)) que pour les logiciels, liée à la diffusion de documents par le Web.

Les licences (en particulier les licences libres) sont utiles pour simplifier et clarifier le cadre juridique et les droits octroyés qui permettent l'utilisation, la modification et la redistribution d'une oeuvre. Les problèmes que les licences peuvent faire apparaître sont liés à la compatibilité (lorsque plusieurs briques logicielles sont utilisées), à l'héritage ou au non respect des licences.

Les lois sont propres à un pays, et ce qui est légal dans un pays peut être illégal dans un autre. La diffusion des documents et des logiciels en utilisant le web ne connaît pas ces frontières, ce qui complique parfois les aspects légaux associés à cette diffusion.

## Aspects relatifs à la politique scientifique

La politique scientifique fait intervenir des décisions qui relèvent de niveaux différents, en voici un classement :

- (C) les chercheurs, les développeurs, les auteurs, les membres d'un laboratoire
- (L) un laboratoire, sa direction
- (T) les tutelles et autres organismes financeurs de la recherche en France
- (CSI) la communauté scientifique internationale au sens large, ce qui comprend par exemple la politique scientifique européenne, les politiques des revues scientifiques, et autres.

Les points traités dans cette section seront classés en fonction de l'intervention de ces quatre niveaux de décision.

### Définition (L, T)

Ce point **Définition** est traité au début du document, il est classé (L, T). Pour un article, ce sont les laboratoires qui établissent les normes pour déterminer si un article fait partie de sa production scientifique. Cette décision est "validée" par les tutelles lors des évaluations des laboratoires. Pour un logiciel, à cause de la gestion des droits patrimoniaux, cette décision fait aussi intervenir les tutelles ou autres organismes financeurs.

### Signature (C, T)

Un article est signé par ses auteurs. La signature de l'article contient donc les informations sur les auteurs suivant un certain format. On y mentionne leur affiliation : leur laboratoire, leur institution, une adresse. Le format à suivre pour cette signature est décidé par les tutelles d'un laboratoire. La

signature d'un article par un membre d'un laboratoire ou d'une institution est, en général, un objet bien défini.

La ligne "Copyright ou Droits patrimoniaux" d'un logiciel doit mentionner ses propriétaires. Elle peut être facile à déterminer lorsque la liste d'auteurs est bien établie, ainsi que leurs affiliations ; mais parfois elle n'est pas évidente à déterminer. Considérons par exemple un laboratoire avec 3 tutelles. Quelle est la signature du logiciel fait par un doctorant qui est physiquement installé dans les locaux de la tutelle 1 et travaille sous la direction du professeur de la tutelle 2 et en collaboration avec un membre de la tutelle 3 ? Il faut se référer aux conventions entre les tutelles (contrats quadriennaux, ...) définissant le statut du laboratoire. De plus la notion d'"établissement hébergeur" peut être appliquée (voir l'article R. 611-13 du CPI).

La signature d'un logiciel est (à mon avis) un objet mal défini. On peut convenir d'appliquer aux logiciels des normes similaires à la signature d'articles à défaut des règles spécifiques, en associant les logiciels aux laboratoires dans lesquels ils sont développés.

Désormais les informations de licence d'une oeuvre devraient faire partie de la signature de cette oeuvre.

### **Références (L, T)**

En France, les [archives HAL](#) peuvent gérer les références d'articles, qui proviennent souvent d'un dépôt de l'article complet. Il est ainsi possible de récupérer la bibliographie d'un laboratoire.

La [plateforme PLUME](#) publie des fiches descriptives de logiciels produits dans les laboratoires de recherche. Grâce à son mécanisme d'indexation, on peut obtenir la liste des logiciels associés à un laboratoire ou à une institution.

Des sites bibliographiques gèrent des fiches d'articles : une référence complète suivie d'un résumé. L'objectif du projet [RELIER](#) est la publication des [fiches de développements ESR](#) pour décrire succinctement les logiciels produits dans les laboratoires.

### **Liste des articles/logiciels d'un laboratoire (L, T)**

Chaque laboratoire gère et publie sa liste d'articles. Ce document est revu régulièrement et surtout mis à jour lors de l'évaluation du laboratoire.

La liste complète des logiciels d'un laboratoire existe rarement. Parfois il s'agit d'une liste bien gérée en interne et souvent quelques logiciels sont affichés dans le site web du laboratoire. La liste de logiciels d'un laboratoire est en général un objet inconnu.

Le document [Guide laboratoire pour recenser ses développements logiciels](#) est destiné à ceux qui souhaitent référencer des logiciels d'un laboratoire (ou d'une institution).

### **Libre accès (C, L, T, CSI)**

Des politiques d'accès libre (open access en anglais) sont de plus en plus courantes dans la communauté scientifique internationale, des organismes français comme le CNRS, l'INRA, l'INRIA, l'INSERM, et la Conférence des Présidents d'Université ont signé la [Déclaration de Berlin \(nov. 2003\) sur le libre accès à la connaissance](#).

Conformément aux conditions des subventions de l'Union européenne qu'ils reçoivent au titre du 7e PC (programme-cadre pour les activités de recherche, de développement technologique et de démonstration), les chercheurs sont censés déposer le texte intégral de leurs publications dans un référentiel public qui les rendra accessibles dans le monde entier de manière permanente. L'initiative [OpenAire \(Open Access Infrastructure for Research in Europe\)](#), lancée en novembre 2010, offre un

réseau de bases documentaires publiques européennes donnant libre accès en ligne aux connaissances produites par ces scientifiques. En accord avec cette politique, l'ANR incite les chercheurs à intégrer leurs publications dans le système d'archives ouvertes HAL.

La déclaration de Berlin est suivie de la déclaration de Ghent (janv. 2011) qui renforce l'initiative du libre accès à la connaissance vers la création et l'utilisation des données ouvertes, des logiciels libres et des ressources éducatives ouvertes pour assurer l'intégration des quatre sections du développement libre : publications, données, logiciels et éducation dans les programmes de recherche de la Communauté européenne.

Pour les articles, la politique d'accès libre promue par les organismes financeurs de la recherche se met en place dans les revues scientifiques, voir par exemple les revues [Logical Methods in Computer Science](#) ou [Bioinformatics](#) qui affichent une politique de open acces.

La mise en place d'une politique en accord avec ces déclarations pour les logiciels de recherche correspond d'une part à l'adoption de licences libres et d'autre part au lancement des dépôts où les logiciels peuvent être accessibles (ce qui est, de plus, lié aux articles L. 131-2, R. 132-9, R. 132-10 du Code du patrimoine lorsque le logiciel est diffusé sur un support physique).

Sur ce deuxième point je souhaite mentionner que la [plateforme PLUME](#) recueille des informations sur les logiciels mais pas les logiciels eux mêmes. Les forges peuvent remplir ce rôle de diffusion de logiciels, l'initiative [Projet de forge Ens Sup Recherche - le périmètre restant à définir](#) est en cours.

Je l'ai déjà indiqué, les licences libres sont utiles pour clarifier et simplifier le cadre juridique des logiciels de recherche. [L'adoption de politiques de licences \(libres\) par défaut](#) par les tutelles simplifie la gestion et permet aux chercheurs une diffusion rapide de leur oeuvre, sans attendre des délais d'acceptation de licence qui ne seront pas très utiles pour la majorité des logiciels de recherche. Des versions plus évoluées du même logiciel feront peut-être intervenir des projets de collaboration ou auront besoin d'autres modèles de valorisation. L'adoption de [licences libres à copyleft fort](#) donne une garantie de plus pour le libre accès au logiciel et à ses oeuvres dérivées dès que sa diffusion est assurée.

Assurer l'accès libre à la recherche fait intervenir tous les niveaux de décision, d'où le classement (C, L, T, CSI) de ce point.

### **Validation (C, L, T, CSI)**

La publication d'un article fait l'objet généralement d'une procédure de soumission et de validation de contenu ("procédure de referee"). Le document soumis est évalué par des experts du domaine, ils indiquent des corrections si nécessaire et proposent ou non le travail à publication.

Qui valide un logiciel ? Une bonne batterie de tests peut indiquer qu'il fonctionne comme les auteurs l'attendent, mais il peut être peu évident que les résultats produits par un logiciel soient valables d'un point de vue scientifique.

PLUME aborde ce concept de [logiciel validé au sens PLUME](#) et établit des critères de validation basiques : pour contourner la difficulté d'une analyse fine du logiciel, ils sont basés sur l'utilisation du logiciel ou non dans un ou plusieurs sites en environnement de travail réel (en production).

Il arrive que des résultats publiés dans des articles soient obtenus avec des logiciels développés à cette intention. Si les logiciels ne sont pas diffusés, la reproductibilité de la recherche n'est pas garantie (voir par exemple [Reproducible Research](#)). Cela peut mettre en cause la validation d'un article publié sans accès au logiciel associé. Des réponses récentes à cette question sont proposées par [les publications executables](#).

L'amélioration de la validation des résultats de recherche fait intervenir des décisions à tous les niveaux, d'où le classement (C, L, T, CSI) de ce point.

## **Qualité et évaluation (C, L, T, CSI)**

Une des mesures utilisées pour déterminer la qualité de l'article est son indice de citation et autres indicateurs bibliométriques (voir par exemple [Du bon usage de la bibliométrie pour l'évaluation individuelle des chercheurs](#)).

La qualité d'un logiciel peut se mesurer par celle des articles associés au logiciel, mais aussi par les utilisateurs qu'il est capable d'attirer, les collaborations et les contrats qu'il est capable de générer.

Des méthodes de développement logiciel peuvent aider à avoir des logiciels d'une meilleure qualité, mais un logiciel de recherche de bonne qualité scientifique ne veut pas dire que l'on a un logiciel de bonne qualité d'un point de vue technique. Il ne faut pas donc confondre ces deux concepts.

L'évaluation de l'activité scientifique est basée presque exclusivement sur la liste de publications ; il est nécessaire de définir des critères d'évaluation pour les autres réalisations de la recherche comme par exemple les logiciels (voir la page 34 de [Du bon usage de la bibliométrie pour l'évaluation individuelle des chercheurs](#) ou aussi le document [L'évaluation individuelle des chercheurs et des enseignants-chercheurs en sciences exactes et expérimentales](#)), les bases de données, les prototypes matériels, les molécules, ... En bref, pour toute la production scientifique d'un laboratoire différente des publications.

## **Motivation (C, L, T, CSI)**

Quelle est la motivation pour écrire un article ? En général, l'article sert à la diffusion de connaissances, à montrer l'état d'avancement d'un travail de recherche. Le fait d'ajouter un élément à la liste de publications (personnelle, d'équipe) est aussi une bonne motivation.

Quelle est la motivation pour réaliser un logiciel (de recherche) ? En général, le logiciel a été fait pour comprendre un objet scientifique ou mener une étude, pour prouver une théorie, réaliser un calcul, pour obtenir des nouveaux résultats ou valider une publication. Il peut être vu aussi comme un canal de diffusion de connaissances. Mais, d'après mon expérience, le logiciel en lui-même est rarement la motivation principale dans le milieu de la recherche.

Ceci explique en partie le fait que les développeurs de logiciels scientifiques souhaitent rarement diffuser leurs codes : ils considèrent souvent qu'ils sont mal écrits et ils ne souhaitent pas perdre du temps à faire des codes plus structurés ou tout simplement documentés. De plus ces codes sont souvent écrits par des scientifiques qui n'ont pas de formation en développement de logiciels.

Le fait qu'il n'y ait pas une vraie procédure de validation renforce les motivations pour ne pas diffuser les logiciels de recherche.

Ces codes peuvent être mal écrits par des personnes avec peu de moyens (humains) ou peu de formation, mais cela ne devrait pas (à mon avis) être un obstacle pour reconnaître, apprécier et valoriser cette énorme quantité de travail de développement qui se fait dans les laboratoires. Les acteurs des décisions politiques à tous les niveaux devraient agir pour faire évoluer la situation, en particulier pour les logiciels associés aux articles de recherche, puisque leur diffusion est importante pour la validation des articles associés et nécessaire à la reproductibilité de la recherche, à son libre accès.

On comprend alors que si la motivation n'est pas de faire un logiciel, on s'occupera peu ou pas du tout de ses aspects légaux ou de sa diffusion.

## **Objet (C, L, T, CSI)**

Tandis qu'un article est clairement un objet scientifique et que d'autres aspects de transfert de technologie sont gérés par des contrats et autres documents, un logiciel est un objet "3D" : il est à la

fois un objet scientifique mais aussi potentiellement un objet de transfert de technologie, voire un objet industriel (voir [Stratégie de l'INRIA sur le logiciel libre](#)). Ces deux dernières dimensions sont mal connues et peu comprises dans les milieux scientifiques qui ont peu d'interaction avec les services de valorisation. Mais ces dimensions sont là, intrinsèques à l'objet, et il faut agir en conséquence.

### Tableau récapitulatif des aspects relatifs à la politique scientifique

Voici un tableau qui récapitule les 9 points traités :

Article vs. Logiciel : aspects de politique scientifique		
	Article	Logiciel
<b>Définition (L, T)</b>	ok	à définir
<b>Signature (C, T)</b>	ok, définie par les tutelles	à définir (ligne de copyright), associer les laboratoires
<b>Références (L, T)</b>	HAL	PLUME
<b>Liste des oeuvres d'un laboratoire (L, T)</b>	document à jour	document inconnu, PLUME peut être utile
<b>Libre accès (C, L, T, CSI)</b>	politique (+/-) ok, dépôt ok (HAL)	politique (licences) à définir, dépôt à établir
<b>Validation (C, L, T, CSI)</b>	procédure de referee, reproductibilité	à définir validé (au sens PLUME)
<b>Qualité et évaluation (C, L, T, CSI)</b>	nombre de citations	en fonction des articles associés, capacité d'attirer d'utilisateurs, de contrats
<b>Motivation (C, L, T, CSI)</b>	recherche, article	recherche, pas le logiciel
<b>Objet (C, L, T, CSI)</b>	scientifique	3D : scientifique, mais aussi potentiel de transfert de technologie, industriel

Table 3. Les 9 points « aspects de politique scientifique » de comparaison entre articles et logiciels.

Ce tableau nous aide à mieux comprendre les différences dans les modèles de production d'articles et de logiciels dans un laboratoire. Le modèle des articles fonctionne bien, il est bien maîtrisé depuis des décennies. Le seul point rouge que j'ai signalé, la reproductibilité des résultats, est lié au logiciel (non diffusé) associé à un article publié : pour le moment, le rôle de ces logiciels dans la procédure d'acceptation de l'article n'est pas bien défini. Les [publications exécutables](#) proposent des réponses à cette question, l'avenir nous dira si elles sont retenues par la communauté scientifique.

Pour les logiciels, des questions importantes comme la définition d'un logiciel de laboratoire (large ou avec des restrictions, avec ou sans diffusion, ...), sa signature, les conditions de sa diffusion, leur rôle dans l'évaluation de la recherche et autres sont encore à étudier et à faire évoluer.

Il n'y a pas, à ma connaissance, une procédure de validation des logiciels bien établie dans la communauté scientifique, même lorsque ces logiciels sont associés à l'obtention de résultats soumis à publication. Ceci n'aide pas à motiver les chercheurs à une diffusion de ce travail indispensable aujourd'hui pour faire évoluer la recherche.

Comprendre le logiciel en tant qu'objet complexe dans ses dimensions scientifique, de transfert de

technologie et industrielle aidera à étudier et à décider les questions que le recensement complet des logiciels d'un laboratoire pose. Ces dimensions n'auront pas le même poids partout (en fonction souvent du thème scientifique, de l'objectif du logiciel, ...) mais leur analyse est nécessaire pour établir la stratégie scientifique d'un laboratoire.

## Conclusion

On a étudié dans ce document les différences et similitudes entre articles et logiciels de recherche avec le double objectif de mieux comprendre les problèmes juridiques et décisionnels que les développements logiciels posent au sein des laboratoires et d'améliorer leur gestion. Nous avons étudié 19 points, et séparé les aspects légaux, relatifs au droit d'auteur et aux licences, des aspects relatifs à la politique scientifique.

Ces deux aspects sont communs à tout laboratoire avec une production logicielle, indépendamment de leurs thématiques scientifiques. Le fait de soulever ces questions apporte (je l'espère) un début de réponse qui sera à adapter pour chaque laboratoire en fonction de sa stratégie scientifique.

Pour mieux mener l'étude, la définition de logiciels a été restreinte ici aux logiciels associés aux articles de recherche, mais cette définition s'étend facilement à d'autres types de logiciels produits dans les laboratoires et plus largement aux logiciels produits dans la communauté d'enseignement supérieur et de recherche, appelés ici "logiciels académiques".

Il y a beaucoup de développements logiciels réalisés dans les laboratoires et les établissements de recherche, ces logiciels sont souvent peu connus, peu diffusés, peu visibles, peu accessibles ; ils sont parfois diffusés mais pas dans les meilleures conditions (pas de licence, aspects légaux insuffisamment traités). Il est indispensable de mieux considérer les aspects légaux intervenant dans la diffusion de ces logiciels. Par ailleurs, les développeurs dans les laboratoires ont besoin de motivation et de support (technique, légal) pour améliorer les conditions de la diffusion de ces logiciels.

[PLUME](#) et [RELIER](#) peuvent aider les laboratoires à mieux gérer et tirer partie de leurs développements logiciels. Avec les [fiches de développements ESR](#), on peut montrer facilement les logiciels d'un laboratoire ou d'une institution. En associant à ces fiches les [fiches de logiciels validés](#), on donne un statut de "validé (au sens PLUME)" à un logiciel.

Les laboratoires et institutions peuvent, en utilisant PLUME, améliorer la situation des logiciels académiques pour permettre et pour accompagner leur valorisation, pour augmenter leur visibilité et leur capacité à susciter des collaborations. En définissant des politiques et des procédures en matière de logiciels et de logiciels libres, on peut renforcer l'évaluation et la validation de la recherche ainsi que sa reproductibilité, son libre accès.

Les questions juridiques et décisionnelles traitées ici sur les logiciels apparaissent aussi dans d'autres productions de l'activité scientifique différentes des publications : les bases de données, les prototypes matériels, les molécules, ... Le cadre juridique appliqué peut être différent (par exemple le droit sui generis des bases de données), mais des licences libres des logiciels sont déjà utilisées pour des prototypes (voir par exemple [Matériel OpenSource \(OpenHardware\) et la recherche](#)). Les tableaux présentés ici peuvent donc s'élargir à d'autres productions et servir de base de réflexion.

La recherche de demain se construit avec les briques faites aujourd'hui, il est nécessaire que ces briques soient solides du point de vue technique, scientifique et légal.

# Références

## Sur les questions juridiques

- [Code de la propriété intellectuelle \(CPI\) en France](#)
- [Le droit des logiciels](#), par Valérie Hospital (DAJ, CNRS) dans la journée Collaboration public-privé pour la valorisation de logiciels à Ivry-sur-Seine le 15 juin 2011
- Les oeuvres libres, Mélanie Clément-Fontaine, thèse de l'Université de Montpellier 1, 2006.
- [Licence & copyright pour les développements de logiciels libres de laboratoires de recherche](#)
- [Méthode pour diffuser un logiciel de laboratoire : recommandations juridiques et administratives](#)
- [Licences Creative Commons \(CC\)](#)
- [Journées du réseau Médecin, Grenoble, MSH Alpes, 5 et 6 avril 2011 - Mutualiser les bonnes pratiques : quelles réalités et quel avenir pour l'édition scientifique publique ?](#)

## Sur les questions de politique scientifique

- Thème PLUME : [Patrimoine logiciel d'un laboratoire](#)
- Projet RELIER : [REférencer les développements Logiciels Internes de l'Enseignement supérieur et de la Recherche](#)
- [Guide laboratoire pour recenser ses développements logiciels](#)
- [Du bon usage de la bibliométrie pour l'évaluation individuelle des chercheurs](#), Académie des Sciences
- [L'évaluation individuelle des chercheurs et des enseignants-chercheurs en sciences exactes et expérimentales](#), Académie des Sciences
- [Reproducible Research](#)
- [Executable papers](#)
- [Archives HAL](#)
- [ArXiv](#)
- [Déclaration de Berlin sur le libre accès à la connaissance](#)
- [OpenAire \(Open Access Infrastructure for Research in Europe\)](#)
- [Pourquoi diffuser un logiciel développé dans un laboratoire ou une université avec une licence libre ?](#)
- [Projet de forge Ens Sup Recherche - le périmètre restant à définir](#)
- [Stratégie de l'INRIA sur le logiciel libre](#)
- [Logical Methods in Computer Science](#)
- [Bioinformatics](#)
- [Oxford Open](#)

## Autres documents PLUME

- [Projet PLUME](#)
- [Les différents types de fiches sur le serveur PLUME](#)
- [Quels droits pour les auteurs de logiciels](#)
- [Méthode pour tracer la propriété intellectuelle dans des codes logiciels](#)
- [Collaboration public-privé pour la valorisation de logiciels à Ivry-sur-Seine, 15 juin 2011](#)
- [L'agence de protection des programmes \(APP\) et le registre Inter Deposit Digital Number \(IDDN\)](#)



- ["Les logiciels libres : soumis au droit d'auteur, dans un contexte international, une jurisprudence en émergence, des défis à relever" RMLL'09](#)
- [Séminaire "Construire son projet sur du libre : quelles précautions prendre ?](#)
- [Matériel OpenSource \(OpenHardware\) et la recherche](#)

#### **Autres documents**

- [Guide pratique PME : pensez propriété intellectuelle !](#)
- [La Cathédrale et le Bazar](#) par Eric Raymond
- [Publish your computer code: it is good enough](#) par Nick Barnes
- [Computer code: incentives needed](#) par Konrad Hinsen (Nature, Correspondence)
- [Computational science: ...Error - why scientific programming does not compute](#) par Zeeya Merali
- [Academic publishers make Murdoch look like a socialist](#)
- [Nicolas Bourbaki](#)
- [Ordinateur Shalosh B. Ekhad](#)
- [Wikipédia](#)
- [Res derelictae](#)