

Livre Blanc



inno³

La Propriété Intellectuelle dans l'Industrie de l'Open Source

Version 1.0

Paris, le 03/06/2016

Table des matières

Propos introductifs.....	3
1 L'industrie de l'Open Source.....	4
1.1 Le rôle de l'Open Source dans la diffusion de l'Open Source.....	4
1.2 L'écosystème du logiciel libre.....	4
1.2.1 Les acteurs du système du logiciel libre.....	4
1.2.1.1 Les éditeurs open source.....	5
1.2.1.2 Les intégrateurs.....	9
1.2.1.3 Les constructeurs.....	9
1.2.1.4 Les utilisateurs.....	10
1.2.1.5 Les organisations à but lucratif et non lucratif.....	10
1.2.1.6 Les centres de recherche.....	11
1.2.2 Les relations entre le projet et les acteurs du système.....	12
1.2.2.1 Les relations entre un projet et ses utilisateurs.....	12
1.2.2.2 La collaboration avec les contributeurs.....	12
1.2.2.2.1 Les enjeux et problématiques liés à la détention des droits de propriété intellectuelle.....	13
1.2.2.2.2 Les Contributor License Agreement, Copyright Assignment ou de Join Copyright.....	14
1.2.2.3 La collaboration avec d'autres groupements.....	15
1.2.2.4 La collaboration entre sociétés.....	15
1.2.2.4.1 Gérer les risques de fork.....	15
2 Organiser la diffusion (accompagner l'ouverture du code).....	16
2.1.1 Gérer les licences libres.....	16
2.1.1.1 Mesurer les enjeux liés au respect des licences.....	16
2.1.1.2 Définir les besoins.....	17
2.1.1.3 Organiser l'accompagnement.....	18
2.1.1.3.1 La création d'une structure ad hoc.....	18
2.1.1.3.2 Gérer les contributions au sein de la société.....	19
2.1.1.3.3 Faire travailler ensemble les différentes équipes.....	20
2.1.1.3.4 S'équiper avec de bons outils.....	20
2.1.2 Gérer les autres aspects juridiques.....	21
2.1.2.1 Structurer le projet.....	21
2.1.2.2 Les politiques en matière de marques.....	22
2.1.2.3 Politique en matière de brevets.....	22
Licence.....	24
À propos d'Inno ³	24
À propos de l'Auteur.....	24

Propos introductifs

L'industrie de l'Open Source n'est plus le rêve de demain, mais la réalité d'aujourd'hui.

Plus aucun développement n'est aujourd'hui réalisé sans qu'il ne soit fait usage de logiciels libres et Open Source. Cela quels que soient les motifs : économiques (l'absence de coût de licence), pragmatiques (l'existence d'une solution existante), ou encore de qualité (la solution open Source étant simplement la meilleure dans son domaine).

La propriété intellectuelle joue un rôle important car elle a justement été créée pour ce type d'utilisateur. L'enjeu, pour les organisations d'aujourd'hui, est donc de savoir comment optimiser l'usage de sa propre propriété intellectuelle et comment se prémunir et respecter (not. en termes de brevets) celles des tiers. À cet égard, les droits de propriété intellectuelle ne sont que des outils, des moyens nécessaires à la réalisation d'un objectif que se fixe l'organisation.

Les différents acteurs organisent ainsi de nouvelles chaînes de valeur qui s'appuient sur la diffusion d'un nombre croissant de ressources sous licences Open Source qui participent d'une logique de partage sauf exception : l'exploitation du logiciel est permise par principe, à condition de respecter les limitations clairement énumérées par les licences (sous peine d'être sanctionnés par une action en contrefaçon). Les contrats et autres droits de propriété intellectuelle (notamment les marques) servent ensuite à peaufiner le modèle de chacun des acteurs à délimiter le périmètre d'ouverture de leur innovation.

Première opus d'une étude qui évoluera au fil des ans, ce livre blanc présente les différents acteurs de l'écosystème de l'Open Source ainsi que les relations construites sur la base des droits de propriété intellectuelle.

1 L'industrie de l'Open Source

Le courant des licences libres, malgré sa jeunesse et sa relative complexité, a rapidement donné naissance à un nouveau système. Ainsi, c'est l'organisation des acteurs privés qui doit être examinée (1.1) avant de se pencher sur l'organisation relative à la diffusion du logiciel sous licence libre (1.2).

1.1 Le rôle de l'Open Source dans la diffusion de l'Open Source

Les fondations et autres organisations à but non lucratif occupent une place prépondérante dans le système du Libre. Elles peuvent être classées en deux catégories¹ : celles qui concourent directement au développement d'un logiciel et celles qui y concourent indirectement.

Les premières sont productrices de code, de créations (qu'elles soient ou non titulaires de droits) : – organisations avec base de code diversifiée (Linux Foundation, Apache Foundation, Eclipse Foundation, OW2 Consortium, Adullact) ;

- organisations supportant une ligne de produits (Libreoffice, Mozilla.org, VideoLan, Genivi, Open Handset Alliance) ;
- communautés orientées technologie (FreeBSD, Ruby, Rails, Joomla).

1.2 L'écosystème du logiciel libre

Par leur organisation, les acteurs privés ont dessiné un nouveau système dont il convient de présenter les acteurs (1.1.1) et les interactions qui le caractérisent (1.1.2)².

1.2.1 Les acteurs du système du logiciel libre

Les licences libres ne sont que des outils. Le corollaire est que tous les types de personnes (physiques ou morales, privées ou publiques, de tout secteur ou métier) les utilisent, avec une pluralité de motivations, mais au profit de la constitution d'un « pot commun » qui les réunit (chacun étant libre d'y verser comme d'y puiser). Les individus, les associations et autres organismes à but non lucratif ont souvent un rapport au Libre complexe qui mêle philosophie, idéologie, et des arguments plus raisonnés comme le coût ou l'indépendance qu'offrent les solutions logicielles libres. Tandis que les entreprises, neutres par essence³, ainsi que les collectivités et administration adoptent le Libre par opportunité et pragmatisme (les logiciels libres assurant, outre une maîtrise des coûts, les avantages d'une grande flexibilité et d'un développement rapide).

¹ Distinction basée sur les travaux présentés par Jean-Pierre Laisné lors de l'OWF 2011.

² On pourra accessoirement se reporter au Rapport Ernst & Young, *Open Source software in business-critical environments*, 2011 (sur <http://www.ey.com>).

³ Voir notamment Frison-Roche (Marie-Anne) : « *Qu'est-ce qu'agir de façon neutre* », dans *la neutralité dans les systèmes de régulation économique* », Colloque annuel de The Journal of Regulation (JR).

Parmi tous les courants et même s'il n'est peut-être pas le plus connu du grand public (le mouvement de l'Open Data est par exemple « très vendeur »), le secteur le plus avancé et structuré est incontestablement celui du logiciel : 50 % des entreprises interrogées par Gartner⁴. ont adopté les logiciels libres dans le cadre de leur stratégie informatique.

Ainsi, l'industrie du logiciel libre devrait représenter 10 % du marché de demandes en logiciels et services français d'ici 2 ans⁵ (renforçant ainsi la compétitivité de la France en matière de développement de logiciels).

Les développements qui suivent seront donc dédiés au système qui caractérise le logiciel libre. On s'aperçoit que les frontières entre les différents acteurs (éditeurs des logiciels, intégrateurs, constructeurs et utilisateurs) sont beaucoup moins étanches que dans le secteur traditionnel du logiciel : les éditeurs intègrent de plus en plus de composants open source dans leurs logiciels, les intégrateurs peuvent prendre le rôle d'éditeur vis-à-vis de leur client (support de niveau 3, développement de nouvelles fonctionnalités, etc.) et les utilisateurs eux-mêmes se réunissent en « une communauté d'utilisateurs éditrice⁶ ». Néanmoins, si on distingue les entreprises simples utilisatrices de composants open source de celles qui orientent leur modèle économique en rapport avec cet usage, et qu'on se concentre sur ces dernières seulement, on peut distinguer différents rôles : éditeurs (1.1.1.1), intégrateurs (1.1.1.2), constructeurs (1.1.1.3), grands utilisateurs clients (1.1.1.4), fondations et communautés (1.1.1.5) et centres de recherche (1.1.1.6).

1.2.1.1 Les éditeurs open source

Il s'agit ici de l'entreprise distribuant sous licence libre un logiciel qu'elle édite. De multiples adoptions de l'open source coexistent dans le monde de l'édition : en fonction de l'éditeur (et de son histoire), du type de produit édité et de ses utilisateurs. Le choix n'est pas systématique, mais les éditeurs qui souhaitent construire un modèle économique innovant, voire disruptif, ont besoin de s'appuyer sur une communauté, préexistante ou nouvellement créée, qui va jouer le rôle de soutien (renforçant le logiciel, se substituant parfois à la société, et constituant une réserve de clients voire de salariés pour la société). À partir de là, le modèle économique de la société est généralement issu d'un équilibre entre ce qui est cédé et le contrôle, plus ou moins fort, conservé par l'éditeur (par un droit exclusif, sur la feuille de route, etc.). Se concentrant sur les seules pratiques juridiques et sans que la liste soit exhaustive, les principaux mécanismes (librement combinables) sont l'usage du droit des marques, la proposition de licence commerciale complémentaire, la mise à disposition décalée ou encore le contrat.

Le plus souvent, les éditeurs assurent la protection de leur produit par l'usage complémentaire du

4 « Gartner Survey Reveals More than Half of Respondents Have Adopted Open-Source Software Solutions as Part of IT Strategy », Février 2011 (<http://www.gartner.com>).

5 Etude OPIIEC Mars 2009 : « Les métiers du logiciel libre en France » (www.syntec-informatique.fr — tendance 2008-2009 du logiciel libre, www.ob2l.com). Voir aussi le Rapport de la Commission pour la libération de la croissance française, sous la présidence de Jacques Attali, La documentation française, 2008 (<http://lesrapports.ladocumentationfrancaise.fr>).

6 Voir Elie (François), Economie du logiciel libre, Paris, Eyrolles, 2009.

droit des marques⁷ : le logiciel est librement diffusé sous licence open source, mais tout usage de la marque est soumis à autorisation. Cette faculté qu'offre le cumul de droits exclusifs concurrents sur un même objet a ici l'avantage de rendre quasi inexistant le coût d'entrée pour les utilisateurs, tout en assurant à l'éditeur principal un contrôle sur l'évolution, la distribution des copies et l'exploitation globale du logiciel. Dans l'usage de ce droit, il est possible de distinguer les sociétés commerciales qui ont tendance à en user de façon très agressive — au point parfois de réduire les libertés concédées par la licence open source sur le logiciel⁸ — tandis que les structures non commerciales en font un usage plus modéré, équilibré, qui a principalement pour objectif de défendre l'image associée au logiciel⁹. Le risque inhérent à ce type d'utilisation du droit des marques est que l'éditeur, trop mercantile, se mette à dos la communauté qui s'émancipe alors en enlevant toute allusion à la marque du logiciel et en reprenant à son compte les logiciels diffusés sous licence open source¹⁰. Aux États-Unis, les brevets sont aussi utilisés comme un levier supplémentaire permettant de convaincre les investisseurs et les clients (Novell en étant le « meilleur » exemple). On observe le même phénomène en Europe, même si l'utilisation des brevets parallèlement à une licence libre reste heureusement un comportement isolé et très critiquable.

Par ailleurs, il est fréquent que les éditeurs ne fassent pas le choix exclusif de l'open source et développent de façon périphérique des offres commerciales en associant à la version libre des add-ons ou plug-ins (greffon) sous licence commerciale¹¹. Dans le même esprit, il est possible de proposer le même logiciel sous une licence commerciale présentée comme alternative à la licence open source¹² ou de proposer un service autour d'une version non encore distribuée du logiciel (à noter néanmoins que ces deux solutions sont conditionnées au fait que la société détienne tous les droits sur le logiciel¹³ ou que les licences des composants open source utilisés n'interdisent pas de

7 Cf. supra concernant l'étude du droit des marques.

8 Un très bon exemple est la société américaine Red Hat : elle n'autorise aucun usage de ses marques, sauf : pour un usage non commercial par des établissements éducatifs, des associations à but non lucratif, aux groupes d'utilisateurs et leurs personnels correspondants ; pour un usage personnel ou professionnel issu d'une version originale sous la condition de non-redistribution. . . Mandriva propose, elle, une licence de marque pour la distribution d'une version non modifiée d'un produit Mandriva lorsque celui-ci est disponible gratuitement sur son site internet.

9 La Mozilla Foundation limite par exemple l'usage des marques Firefox et Thunderbird selon la version du logiciel (licence complète sur la version binaire non modifiée, licence large pour les versions officielles localisées, licence réduite pour la version communautaire et enfin aucune licence pour une version du logiciel trop modifiée).

10 Red Hat a ainsi vu naître CentOS, version « expurgée » de la première (opération qui est par ailleurs techniquement compliquée par Red Hat), et Firefox s'est vu opposer Iceweasel (avec un succès plus mitigé) — aujourd'hui GNU IceCat.

11 Cette pratique est surtout répandue en matière de distribution ou compilation de logiciel, c'est-à-dire des situations où des briques non libres peuvent être mêlées à d'autres qui le sont. C'est notamment le cas de Red Hat et Mandriva.

12 Encore faut-il que cette licence soit suffisamment contraignante pour intéresser au choix de la licence commerciale. Ce sont les exemples de MySQL AB, Sleepycat Software Inc., TrollTech AS, etc. Ainsi, MySQL AB qui propose MySQL Entreprise sous licence GNU GPL v 2 (avec exception en faveur d'autres logiciels libres) ou sous licence propriétaire pour ceux qui ne désirent pas respecter la licence (notamment pour développer ou distribuer un projet propriétaire) ou veulent bénéficier de mises à jour, de service support et assurance.

13 Ce dont MySQL s'assure en n'intégrant dans le produit que les contributions qu'elle a pu réécrire ou racheter.

telles utilisations ou distributions fermées). Dans ce second cas, il est possible de penser à un modèle du type free cloud telle que l'offre Drupal Garden de la société Acquia : un ou plusieurs services fournis sur la plate-forme de l'éditeur (par exemple création et hébergement d'un site) s'harmonisent avec la distribution open source du logiciel puisque l'utilisateur peut à tout moment exporter le code et le contenu générés pour les installer sur sa propre instance. Dans ces hypothèses, mixer licences open source et commerciales¹⁴ permet divers montages dans lesquels les communautés continuent à trouver leur intérêt à utiliser et collaborer au développement du logiciel grâce à la licence open source, tandis que les utilisateurs professionnels préfèrent à terme opter pour une licence commerciale¹⁵.

De façon assez similaire, certaines sociétés proposent des mises à jour plus simples et régulières à leurs clients (non immédiatement accessible aux utilisateurs de la version communautaire) : soit par une mise à disposition systématiquement décalée (de quelques mois ou quelques semaines) ; soit par une mise à disposition graduelle des mises à jour pour les clients (alors qu'elles seraient regroupées pour la version complémentaire). Ce décalage peut être matériel ou juridique (les mises à jour ne sont considérées comme divulguées qu'à compter d'une certaine période).

Enfin, certaines sociétés usent du droit des contrats afin de limiter accessoirement l'utilisateur dans les libertés que lui concède la licence open source : que ce soit lors de la vente de services aux utilisateurs qui souhaitent bénéficier du support et des mises à jour¹⁶ ou dans les rapports entre distributeurs et revendeurs¹⁷. Les contrats sont aussi essentiels pour les logiciels orientés métiers demandant de concevoir plus finement la collaboration entre les différents industriels¹⁸ : la licence open source n'étant alors plus qu'une partie du cadre contractuel (s'ils sont concurrents, une licence copyleft sera une condition *sine qua non* afin d'assurer une relative pérennité et équité dans le développement du logiciel) — un contrat de consortium complétant éventuellement tous les aspects non couverts par la licence (comité de direction, procédé de développement, prise en compte des orientations, marketing et communication, etc.)

Par exemple...

14 Le terme peut sembler ambigu : une licence open source autorise parfaitement un usage commercial du logiciel, mais elle n'en restera pas moins une cession non exclusive gracieuse ; à l'inverse, une licence dite commerciale sera conditionnée au paiement d'une souscription quelconque.

15 A la différence de la licence open source par ailleurs proposée (probablement copyleft ou tout du moins suffisamment contraignante pour que l'utilisateur ait un intérêt à opter pour la licence commerciale), la licence commerciale de l'éditeur offre aux clients des avantages comme la possibilité de ne pas diffuser le contenu de leur création dérivée (échappant ainsi à la réciprocité afin de continuer de disposer de leur monopole).

16 C'est notamment le cas de Red Hat avec les produits comme JBoss Enterprise Mid- dleware, Red Hat Directory Server, Red Hat Certificate System, etc.

17 On retrouve ce type de limitation dans les contrats de MySQL, Ingres, Alfresco, etc. Le revendeur se retrouvant bien souvent contraint de ne pas utiliser ou conseiller la version communautaire.

18 La licence open source assurera la confiance entre ces derniers vis-à-vis de ce qui est créé et permettra éventuellement à d'autres acteurs de bénéficier d'un coût d'entrée quasi inexistant — de façon à ce qu'ils puissent par la suite éventuellement rejoindre le cercle des partenaires ; par ailleurs, le consortium jouera le rôle d'un contrat d'indivision par lequel les partenaires s'organisent pour la gestion du projet (en amont lors de la création et en aval pour l'exploitation du projet).

Révéléateur de l'intérêt de l'open source, les diverses structures représentatives de l'industrie du logiciel se positionnent progressivement sur le sujet et mettent en place des commissions open source. Ainsi, Syntec Numérique a rédigé un Guide Open Source¹⁹ et créé un comité open source avec pour mission de « fixer le cadre contractuel et d'usage en matière de projets informatiques incluant du logiciel libre : comment s'inscrit le logiciel libre dans l'architecture des projets informatiques, en matière technique, juridique, business model, organisationnelle et distribution ». À son tour, l'AFDEL qui avait abordé les logiciels libres au travers d'un guide²⁰, a récemment mis en place une commission dédiée, présidée par Bertrand Diard (Talend), avec pour mission de « réunir et fédérer les éditeurs de logiciels en France pour débattre de thèmes spécifiques liés au modèle open source, et rédiger des conclusions et propositions sur ces thèmes ». Le Cigref a aussi récemment publié son guide intitulé « Maturité et gouvernance de l'Open Source : la vision des grandes entreprises » (<http://www.cigref.fr>).

MICROSOFT ET L'OPEN SOURCE

Le comportement de cette société peut, à divers égards, sembler paradoxal.

– Microsoft a ouvertement qualifié le mouvement du logiciel libre de « cancer »²¹ avant de rédiger ses propres licences labellisées open source (Microsoft Public License (Ms-PL) et Microsoft Reciprocal License (Ms-RL)) au sein du programme Shared Source Initiative (initiative destinée à favoriser l'innovation technique, elle permet un accès favorisé au code source des logiciels de la société au profit des clients, partenaires, chercheurs et développeurs). – Un laboratoire open source a été mis en place, et Microsoft s'enorgueillit de contribuer à plusieurs projets libres (voir le billet intitulé « Microsoft Contributes Linux Drivers to Linux Community », sur Microsoft.com).

– La société a une politique en matière de brevets généralement contraire à l'open source : elle oppose régulièrement ses brevets contre des projets open source ; et a racheté des brevets de Novell par l'intermédiaire d'un consortium (CPTN Holdings LLC) qu'elle dirige ; même si elle a par ailleurs publié deux promesses de non-agression (« Patent Pledge for Open Source Developers » ; « Microsoft Approach to Interoperability »).

– La société s'est rapprochée de l'éditeur Novell au sein d'un Interop Ability Lab (interopérabilité entre Windows serveur et Suse Linux Entreprise : <http://www.moreinterop.com>).

– La WebApp de Microsoft comporte par ailleurs des clauses interdisant aux développeurs d'utiliser directement ou indirectement des composants sous licence open source (voir Computing.co.uk, « Microsoft restricts use of open source in app ») : « the application must not include software, documentation, or other materials that, in whole or in part, are governed by or subject to an 'excluded license', or that would otherwise cause the application to be

19 Disponible sur : <http://guideopensource.info>. Voir J EAN (Benjamin) et F LIPO (Olivia), « Bonnes pratiques en matière de conception et de pilotage de projets informatiques incluant ou composés de briques logicielles libres », EOLE 2009.

20 Voir notamment l'étude « Propriété intellectuelle et logiciels, la position de l'association Française des éditeurs de logiciels », AFDEL, 2007.

21 Voir G REEN (Thomas C.), « Ballmer : 'Linux is a cancer' ; Contaminates all other software with Hippie GPL rubbish » (The Register, 2001), qui cite les propos de Steve Ballmer lors d'une interview pour le Chicago Sun-Times : « Linux is a cancer that attaches itself in an intellectual property sense to everything it touches. »

subject to the terms of an excluded license. ». Sont ainsi exclues les licences GNU GPL et AGPL, et équivalentes. . .

Depuis peu, la société communique sur son modèle économique intitulé Open Surface (un cloud qui mêle logiciels libres et propriétaires).

1.2.1.2 Les intégrateurs

De leur côté, les intégrateurs de logiciels libres et open source (ou de solutions partiellement composées de ceux-ci) subissent généralement plus les licences qu'ils ne les choisissent : leur seule préoccupation étant que les briques logicielles puissent aisément se combiner entre elles. Il n'y a donc pas d'usage du droit des marques à l'égard des produits, mais surtout une expertise approfondie des combinaisons de licences leur permettant de s'assurer de la faisabilité du projet et de s'engager sur ce point auprès de leurs clients et dans leurs contrats (les clients demandant aujourd'hui la garantie de telle compétence ainsi que la livraison de certaines études précises sur ces questions). Transférant généralement un logiciel qu'il n'a pas (à priori) lui-même développé, mais seulement adapté au besoin de son client, l'intégrateur est lui-même limité par sa qualité de licencié à l'égard des auteurs des logiciels libres sur lesquels il intervient, et, ne pouvant céder plus de droits qu'il n'en possède lui-même, il communiquera à ses clients cesdits logiciels sous leur licence open source d'origine (ou toute autre licence compatible).

En ce qui concerne leurs contributions spécifiques, les pratiques sont multiples, mais il semblerait finalement que le plus avantageux pour l'ensemble des parties soit que la titularité des droits soit conservée par celui qui maintient et édite le logiciel (généralement l'intégrateur). Ainsi celui-ci cède l'ensemble de ses développements sous licence libre au client et conserve la titularité des droits. Ce compromis reste fidèle au procédé de création open source dans lequel l'auteur reste titulaire de droits et cède sous licence open source de façon non exclusive, afin de permettre tant au client de mutualiser les résultats de la prestation, qu'à l'intégrateur de valoriser les travaux par tous moyens appropriés²².

1.2.1.3 Les constructeurs

Concourant à la création d'une industrie réunissant innovations matérielles et logicielles, les constructeurs (tels HP, Intel, IBM ou Dell) embarquent de plus en plus de logiciels et se rapprochent à ce titre des différents projets de logiciels libres (et l'open cloud) — devenant de jour en jour des contributeurs majeurs dans les projets OS (Linux, Busybox, etc.) au point de faire de leurs contributions respectives des éléments différenciant.

En effet, de tels logiciels permettent, en plus d'écorner certains monopoles, de réduire le coût de développement, d'optimiser l'interaction entre les logiciels et leurs matériels (ou a minima

22 L'intégrateur se révèle être le mieux disposé pour faire évoluer le produit et assurer la pérennité de son développement au travers, notamment, d'un reversement systématique au bénéfice des communautés. Sa situation d'intermédiaire lui permet par ailleurs d'être idéalement placé pour favoriser les échanges indirects de contributions entre ses différents clients — l'évolution des solutions étant aussi dans l'intérêt de ceux-ci, tout le monde profite de cette collaboration.

d'assurer la compatibilité avec leur produit).

À l'instar des intégrateurs, les constructeurs sont généralement désireux d'une compatibilité maximum entre les licences des logiciels open source utilisés, même s'ils peuvent avoir intérêt à définir certaines politiques : favorables à certaines licences permissives (par exemple dans le secteur du mobile, avec la diffusion d'Android sous licence Apache) ; ou exemptes de certaines obligations (par exemple, de licences de brevets) qui seraient incompatibles avec leur politique plus vaste relative à la propriété intellectuelle.

De nombreux secteurs sont concernés par une certification (ou qualification, cf. supra) de leur produit : l'open source est pour eux une solution qui permet de mutualiser les frais de développement tout en conservant le bénéfice personnel de la certification de leur version.

1.2.1.4 Les utilisateurs

Contrairement aux logiciels traditionnels, les logiciels libres sont généralement accessibles à toutes catégories d'utilisateurs, dans les mêmes conditions. Ainsi, libre à chacun (individu, société, organisme à but non lucratif, collectivité ou État) de s'installer un forum (Phpbb), un wiki (Mediawiki), voire un Facebook-like (Elgg) ou un Youtube-like (Mediacore), etc.

Les utilisateurs d'un projet open source sont particulièrement sensibles à la licence du projet : l'impact de cette dernière étant fonction de l'utilisation qu'ils en font (par exemple, si des services sont proposés sur la base de ces logiciels libres) voire de leur forme (société, GIE, association, etc.).

Lorsqu'il s'agit de sociétés « utilisatrices poussées », elles peuvent faire appel à des intégrateurs ou consultants pour mettre en place les solutions open source dans leur SI — ce qui les place dans une situation analogue aux utilisateurs standards, avec quelques spécificités relatives aux contrats passés avec ces prestataires. En l'absence de tels intermédiaires, il est aussi possible de se tourner vers des solutions d'assurance (telle l'offre américaine Open Source Risk Management proposée par la société OSRM).

Néanmoins, l'impact de l'utilisation des logiciels libres peut être lié à des politiques stratégiques ou économiques particulières : par exemple en cas d'externalisation quasi totale du support, maintenance, veille open source à partir d'un SI essentiellement composé de logiciels libres, ou, au contraire, lors de la mise en place d'une entité informatique interne qui gère le déploiement de solutions en logiciels libres au sein de toute la structure (avec une vraie logique de service et d'édition). Dans toutes ces situations, les utilisateurs veillent à ce que les développements réalisés pour leur compte soient ensuite mis à disposition du (et si possible intégrés dans) projet open source — gardant à l'esprit qu'il est généralement aussi cher de maintenir un composant logiciel dans la durée que de le faire développer.

1.2.1.5 Les organisations à but lucratif et non lucratif

En plus des acteurs précédents, on retrouve plusieurs types de groupements sur lesquels s'appuient les projets libres et open source : ils sont commerciaux (tels les sociétés ou GIE) ou non

(associations ou fondations), avec une personnalité morale ou non (association de fait— fréquent lors du lancement d'un projet — ou consortium), voire multiples (tel OpenOffice.org qui reposait sur l'association TeamEV, mais dont les droits de PI étaient gérés par Sun puis Oracle).

Les fondations et autres organisations à but non lucratif occupent une place prépondérante dans le système du Libre. Elles peuvent être classées en deux catégories²³ : celles qui concourent directement au développement d'un logiciel et celles qui y concourent indirectement.

Les premières sont productrices de code, de créations (qu'elles soient ou non titulaires de droits) : – organisations avec base de code diversifiée (Linux Foundation, Apache Foundation, Eclipse Foundation, OW2 Consortium, Adullact) ;

- organisations supportant une ligne de produits (Libreoffice, Mozilla.org, VideoLan, Genivi, Open Handset Alliance) ;
- communautés orientées technologie (FreeBSD, Ruby, Rails, Joomla).

Les secondes concourent indirectement au développement de projets (ne produisent pas de code per se) :

- organisations fournissant des ressources organisationnelles (administratif, hébergement, conseil, etc.) aux projets (Software Freedom Conservancy, Eclipse, Linux Foundation) ;
- organisations fournissant des ressources légales (Free Software Foundation, Open Source Initiative, Software Freedom Law Center, VVL) ;
- organisations de standards (W3C, IEEE, ODF Alliance, OSGI Alliance, ETSI) ;

organisations de lobbying et de promotion (April, Aful, Framasoft, Open Source for America, Ubuntu-fr, Fedora Fr, Adullact, etc.)

1.2.1.6 Les centres de recherche

L'implication et l'utilisation des logiciels libres par les centres de recherche sont si importantes qu'il est impossible de ne pas considérer les centres de recherches²⁴ (et, dans une moindre mesure les universités²⁵) comme des acteurs à part entière du système que nous décrivons (notamment pour au regard de leur recherche à application industrielle).

Leur apport est néanmoins limité aux domaines directement liés à leur activité de recherche et ils ne prennent ainsi qu'un rôle passif dans l'animation des communautés ou la proposition de services. Ainsi, et même si la situation semble évoluer, on s'aperçoit que leur action se limite aujourd'hui au stade qui précède la valorisation de leurs projets Open Source : cette dernière phase étant réalisée par des sociétés existantes, partenaires ou non, ou nouvellement créées (le transfert en faveur de

23 Distinction basée sur les travaux présentés par Jean-Pierre Laisné lors de l'OWF 2011.

24 L'INRIA et le CNRS étant les deux centres français les plus actifs sur le sujet.

25 L'Université de technologie de Compiègne (UTC) est ainsi à l'origine de nombreux projets Open Source dans le domaine du logiciel (scenari-platform pour n'en nommer qu'un) et en dehors.

spin off se multiplie).

1.2.2 Les relations entre le projet et les acteurs du système

Enfin, il est fréquent que l'intégrateur soit tenu de contribuer aux projets open source en fournissant les modifications réalisées pour le client : il s'agit là d'une sécurité tant pour le client qui s'assure ainsi que les versions ultérieures du projet intégreront ces développements (et qu'il n'aura ainsi pas à le payer une seconde fois) que pour l'intégrateur qui ne souhaite généralement pas endosser la responsabilité d'un éditeur.

Chaque relation mérite néanmoins un examen dédié : entre un projet et ses utilisateurs (1.1.2.1), entre un projet et ses contributeurs (1.1.2.2), entre le projet et les autres groupements (1.1.2.3) et entre les sociétés (1.1.2.4).

1.2.2.1 Les relations entre un projet et ses utilisateurs

Les utilisateurs sont les premiers destinataires des logiciels. Ils contribuent généralement en livrant des retours sur leur expérience. Ils sont aussi souvent les premiers générateurs de valeur, le nombre d'utilisateurs d'un logiciel étant généralement un facteur multipliant la valeur du logiciel vis-à-vis de ceux qui ne l'utilisent pas encore. **La licence utilisée sur l'œuvre logicielle est souvent le seul document les intéressant et, au sein de ce dernier, seuls les droits d'utilisation les concernent véritablement.**

Dans le cadre de certains projets, il arrive que des limitations s'ajoutent, en raison, par exemple, de composants commerciaux : l'utilisateur y consentira alors lors du téléchargement, de l'exécution ou de l'installation du logiciel.

Ces utilisateurs peuvent néanmoins devenir clients, contributeurs et/ou partenaires potentiels, voire concurrents du projet, et il est toujours utile de prévoir à leur attention des documents les informant de leurs droits et devoirs au terme de la licence, des conditions d'utilisation des marques, logos et noms du projet, voire, s'il y a lieu, du processus leur permettant de contribuer au projet (ainsi que la politique du projet en matière de propriété intellectuelle).

EXEMPLE PRATIQUE

Par application d'un effet de réseau, les logiciels libres gagnent en intérêt et fonctionnalité au fur et à mesure que la communauté d'utilisateurs (et de développeurs) s'en approche. Ainsi, les utilisateurs de téléphones Android ont formé des projets, tel le projet CyanogenMod, qui distribue des ROM optimisées pour de nombreux portables, permettant de bénéficier des dernières versions d'Android sur les téléphones qui ne sont plus maintenus par leur constructeur. Cela grâce à la version communautaire du système d'exploitation Android qui est publiée par Google sous licence Apache (et sans embarquer les applications propriétaires telles google maps, voice, search, etc.).

1.2.2.2 La collaboration avec les contributeurs

Les caractères évolutifs et participatifs étant l'essence même du logiciel libre, il n'est pas exclu que

certain contributeurs souhaitent apporter leur lot de développements (envisagés sur un mode projet, en constante amélioration). Dans la situation d'un éditeur (ou la communauté qui revêt ce rôle) déjà identifié sur le projet, il peut proposer plusieurs schémas de contribution qui conditionneront la reprise du code au sein du projet :

- lorsque ces développements concernent le logiciel lui-même, il est souvent demandé une cession globale (ou suffisante) des droits (il est aussi fréquent, au sein des projets commerciaux, que les sociétés rachètent ou réécrivent les contributions qui les intéressent) ;
- lorsque ces développements portent sur des modules communautaires, les solutions sont généralement liées au choix initial de la licence du projet (mais il est aussi possible que l'éditeur n'héberge ou ne maintienne la contribution qu'en cas de cession de droits ou d'une politique de licence précise).

Dès les premières contributions au projet, il devient nécessaire d'avoir une réflexion pointue sur la gestion de la propriété intellectuelle, donnant éventuellement naissance à une politique en la matière, que celle-ci soit très protectionniste ou au contraire très ouverte.

1.2.2.2.1 Les enjeux et problématiques liés à la détention des droits de propriété intellectuelle

La question de la détention des droits de propriété intellectuelle, de la « titularité », est d'une importance variable dans un projet open source. En fonction de la licence (surtout si cette dernière est copyleft) et du type de projets, la détention peut être cruciale en ce qu'elle confère un certain contrôle sur le projet (par exemple pour toutes les sociétés qui pratiquent les doubles licences, dit dual licensing). Cette détention peut au contraire être très faible lorsque la licence choisie est très ouverte, lorsque le projet est soumis aux licences de composants open source qu'il intègre, ou encore lorsque trop de contributeurs se partagent cette titularité.

Pour remédier à cette division de la propriété, de nombreux projets recourent à des politiques de licences spécifiques, de cession de droits ou de « copropriété libre » (chaque copropriétaire pouvant exploiter librement les droits détenus) : par le biais de Contributor License Agreements, Copyright Assignment ou de Joint Copyright qui conditionnent généralement la possibilité pour un développeur de voir sa contribution être acceptée dans un projet ou, dans le cadre professionnel, pour un client qui souhaite que ses développements soient maintenus par le projet. C'est une politique que l'on retrouve à l'égard des salariés d'une société — qui sont aujourd'hui les principaux contributeurs, sur tout type de contenu²⁶.

Inversement, afin de se prémunir face à un éventuel contrôle sur un logiciel libre du fait de la détention des droits par un ou plusieurs, certains peuvent chercher à bloquer l'exercice de ces prérogatives en imposant une convention de copropriété (ou d'indivision) qui lie les parties quant à

²⁶ Pour rappel, la dévolution légale des droits en faveur de l'employeur ne joue qu'en présence d'une création salariale logicielle. Voir notamment BENSAMOUN (Alexandra), « la titularité des droits patrimoniaux sur une création salariée : du paradis artificiel à l'artifice du paradis », dans Revue Lamy Droit de l'Immatériel, n°67, 2011.

l'exploitation conformément à la licence open source du projet²⁷.

1.2.2.2.2 Les Contributor License Agreement, Copyright Assignment ou de Join Copyright

Il est ainsi fréquent que le nombre de contributeurs (et donc de titulaires de droits) dépasse la dizaine voire le millier. Leur anonymat ainsi que leur dispersion géographique sont parfois des obstacles infranchissables pour la suite du projet : l'absence de consentement d'un seul interdit toute décision concernant la création dans son ensemble et la seule échappatoire reste alors de supprimer les contributions pour lesquelles il manque cette autorisation.

C'est pour cette raison que, très tôt, certains développeurs ou organismes, telle la FSF, ont mis en place des documents qui permettent aux contributeurs d'un projet soit de leur céder leur copyright — pour centraliser la prise de décision (notamment le changement de licence²⁸), la gestion, et la défense de l'œuvre — soit de leur consentir une licence très large. On parle de Copyright Assignment en présence d'un vrai transfert (cession exclusive) de droits, la structure devenant la seule et unique titulaire de droit. On parle de Contributor License Agreement lorsque le contrat opère une cession (non exclusive) très large au profit de la structure sans que le contributeur soit lui-même dessaisi de ses prérogatives.

Ils sont généralement doublés d'un second document à destination des employeurs lorsque les contributions sont réalisées dans un cadre salarial. Chacun de ces contrats est rédigé en fonction de l'équilibre particulier du projet, de l'implication des contributeurs dans celui-ci et de la confiance dont bénéficie l'organisme collecteur.

Devant le constat de la multiplicité, de la diversité et de la complexité de ces documents, de nombreux industriels, professionnels et associations du libre ont formé un groupe de travail pour plancher sur la rédaction de documents types à destination des communautés et entreprises : leur épargnant un travail d'écriture en même temps qu'ils épargnent à leurs utilisateurs-contributeurs un travail de lecture (et de compréhension). Le projet Harmony²⁹, s'il permet de mutualiser les politiques auparavant éclatées³⁰, ne résout néanmoins pas la difficulté du rapport que cela génère entre les parties, les uns voulant un engagement fort des contributeurs — notamment par le biais de clauses de responsabilité et de garantie —, les autres voulant un engagement fort des collecteurs — notamment quant à l'exploitation des contributions.

Par ailleurs, ces contrats ne sont réellement fiables qu'en cas de certitude quant à l'identification du contributeur. C'est la raison pour laquelle la Free Software Foundation (mais Mozilla aussi) envoie un contrat papier qu'il convient de retourner signé. De même, l'Eclipse Foundation a mis en place

27 C'est notamment le choix pris pour l'ENT Lillie commandée par la Région Île-de-France.

28 Plusieurs situations de blocage peuvent être prises comme exemple : le changement de licence du contenu du site de Debian au profit d'une licence compatible avec la GNU GPL, l'ajout des licences GNU LGPL, et sur le navigateur Firefox (en faisant une triple licence MPL/LGPL/GPL) ou encore le noyau Linux qui utilise actuellement volontairement la seule version 2 de la GNU GPL.

29 <http://harmonyagreements.org>

30 Voir M. EEKS (Michael), « Some brief thoughts on Project Harmony », 2011 (disponible sur <http://people.gnome.org>).

un processus complet visant à sécuriser les relations entre les différents acteurs (Eclipse Legal Process³¹).

1.2.2.3 La collaboration avec d'autres groupements

Il est très fréquent que les différents projets ou organisations se réunissent pour travailler de concert sur des projets communs : soit qu'ils aient des compétences ou expériences complémentaires, transverses (tels les aspects juridiques liés aux licences et aux brevets), soit qu'ils mettent en commun leurs forces pour le développement d'une solution plus vaste.

1.2.2.4 La collaboration entre sociétés

Une société éditrice de logiciels qui met en place son modèle commercial prend généralement en considération ses partenaires, ses concurrents et ses clients. Il est assez simple de séparer les uns des autres : la société est liée par contrat avec ses partenaires (distributeurs, revendeurs, etc.) et ses clients (contrats souvent négociés pour les premiers et imposés pour les seconds), et n'entretient que des relations contractuelles exceptionnelles avec ses concurrents.

Les licences libres modifient cet ordre puisqu'elles s'adressent autant aux individus — personnes physiques — qu'aux entreprises commerciales. Ainsi, les concurrents comme les partenaires pourront en revendiquer le bénéfice, leur permettant notamment d'utiliser ou de proposer des services autour des logiciels édités par un autre avec pour seule limite le respect de la licence libre du projet et éventuellement de la politique de marques ou de brevets.

On peut y voir un paradoxe puisque les partenaires risquent d'être « moins libres » que les concurrents, du fait de clauses supprimant le bénéfice de la licence, qui peuvent être comprises dans leurs engagements avec l'éditeur³².

1.2.2.4.1 Gérer les risques de fork

On parle de fork lorsqu'une partie d'une communauté décide de s'éloigner du projet principal pour en développer une version différente.

De manière générale, les forks préjudicient à tous : l'ancienne et la nouvelle communautés, les utilisateurs et les projets.

C'est la raison pour laquelle il est généralement conseillé d'augmenter l'investissement dans un projet plutôt que d'initier son propre projet³³.

DE STAR OFFICE À LIBRE OFFICE

Récemment, la communauté d'OpenOffice.org s'est détachée du projet pour proposer sa propre version (nommée LibreOffice) au sein d'une fondation créée à cet effet : la Document

31 <http://www.eclipse.org/legal/EclipseLegalProcessPoster.pdf>.

32 Clauses d'usage dans les contrats de partenariats proposés par MySQL, Ingres, Talend, etc.

33 Voir notamment la présentation de POUTY (Tim), « Becoming a Samba Developer », Isilon Systems, 2009 (disponible sur <http://sambaxp.org>).

Foundation. Cette dernière abrite le code de LibreOffice, mais aussi de tout autre produit qui souhaiterait dériver de ses sources. Lors de la formation du projet OpenOffice.org en 2000, la mise en place d'une telle fondation avait été annoncée par Sun. Dix ans après, et pour se protéger de la nouvelle politique commerciale imposée par Oracle (qui a racheté Sun) sur ses produits, les principaux responsables communautaires du projet ont pensé qu'il était temps de monter une fondation vraiment indépendante. Il n'y a pas de rupture avec le projet, simplement une communauté suffisamment mature pour prendre son destin en main et ne plus dépendre d'un décideur unique. LibreOffice (ou LibO) est une suite bureautique, développée par la fondation, qui s'appuie sur les sources d'OpenOffice.org et contient des patches fournis par ses contributeurs (les plus assidus étant Novell, Red Hat et Debian, mais il y en a de plus en plus chaque jour).

Quelques mois après ces événements, et pour répondre aux fortes demandes d'IBM, Oracle a « donné » son code et ses marques à la Fondation Apache qui distribue ainsi OpenOffice.org sous licence Apache.

Il faudrait certainement se projeter dans quelques années pour mesurer l'impact de ce fork (sur l'entreprise, les communautés, voire, plus globalement, le projet global), mais beaucoup y voient une excellente nouvelle et de nombreux concurrents d'Oracle commencent à retrouver un intérêt dans le développement de cette version dont le succès dépendra de tous et non plus majoritairement d'une société (ne serait-ce que par la détention du patrimoine immatériel). Gageons aussi que cette « insurrection » sera perçue comme une épée de Damoclès pour toute société qui chercherait à trop contrôler un projet qui se veut ouvert. Espérons aussi que les deux projets parviennent à mutualiser leurs efforts.

2 Organiser la diffusion (accompagner l'ouverture du code)

L'organisation de la diffusion est souvent la clé qui assure la pérennité d'un projet, sa stagnation ou son abandon. Malheureusement, il n'y a pas une organisation idéale, une structure parfaite, mais uniquement des cas d'espèce. Pour une entreprise, cela implique de se reposer sur un ensemble hétérogène de personnes au sein de l'entreprise qui perçoivent les enjeux de l'open source et qui s'assure du respect des licences.

Ainsi, que ce soit pour ce seul projet ou pour une multitude, il est nécessaire d'entamer une telle réflexion et, une fois les besoins identifiés, d'organiser l'accompagnement — que ce soit en interne ou en externe³⁴.

2.1.1 Gérer les licences libres

La mise en place d'une bonne gestion des licences libres passe nécessairement par une étude des enjeux pour la société et/ou le projet (1.2.1.1), la définition des besoins (1.2.1.2) et l'organisation de l'accompagnement (1.2.1.3).

2.1.1.1 Mesurer les enjeux liés au respect des licences

Quelles sont les sanctions juridiques, mais aussi commerciales ou marketing, susceptibles de

³⁴ Voir le Guide Open Source, op. cit. (<http://guideopensource.info>).

découler du non-respect des licences libres ? De nombreuses sociétés, pourtant importantes, ont ainsi dû baisser l'échine face aux actions des communautés qui leur reprochaient l'absence de prise en compte des licences sur des composants utilisés (ce qui inclut BestBuy, Westinghouse et Cisco). Ajoutons que les titulaires de droits, généralement seuls aptes à faire respecter leur licence, ont tendance à les accompagner l'utilisation de leur logiciel open source).

En effet, tout non-respect de licence emporte le risque de se voir qualifier de contrefacteur : tout usage n'entrant pas dans le cadre de la licence est un usage non autorisé et donc contrefaisant. Par ailleurs, en cas de violation, par certains usages de droits de propriété intellectuelle, ou plus fréquemment, du non-respect du formalisme de la licence, celle-ci sera généralement automatiquement résiliée et l'ancien licencié se trouvera rapidement contrefacteur pour tout usage de la création. Cela vaut pour l'entreprise, mais aussi pour tous ses clients et partenaires qui redistribuent les logiciels « non conformes » (on perçoit d'ailleurs une telle prise de conscience en Asie).

Au minimum (mais ce n'est pas toujours vrai), il est possible de considérer que lors de l'utilisation d'un composant open source sans modification, la simple redistribution (tel qu'il a été reçu) devrait suffire. Au contraire, une démarche plus lourde serait nécessaire en cas de modification.

La solution la plus fiable est néanmoins de mettre en place un formalisme générique, construit sur le modèle de la licence la plus contraignante, de manière à ce que son respect suffise pour remplir les attentes classiques de l'ensemble des licences. Enfin, des procédures particulières peuvent être élaborées afin de prendre en compte les spécificités des utilisateurs (professionnels du métier ou non, de nationalité précise, etc.) ou du domaine d'exploitation (embarqué, SaaS, etc.).

2.1.1.2 Définir les besoins

En fonction des enjeux, plusieurs besoins peuvent être identifiés :

- la mise en place de bonnes pratiques (guidelines). Il s'agit ici de l'édiction de procédures relatives au respect du formalisme des licences ainsi qu'une procédure associée à l'usage de toutes nouvelles licences/composants open source. Celles-ci permettraient par ailleurs de rendre transparente la gestion des droits de propriété intellectuelle tant à l'intérieur qu'à l'extérieur de la société ;
- la mise en place d'une veille relative aux licences, pratiques afférentes et outils d'audit de codes (il est en effet important de rester connecté aux usages qui se développent, voire d'y confronter ses nouvelles pratiques, afin de se tenir informé tout en conservant un regard critique) ;
- la création d'un support relatif aux questions liées à l'utilisation des licences ou autres questions juridiques. Il s'agit ici des problématiques liées à l'ajout de licences et aux enjeux de compatibilité, mais un support juridique étendu peut permettre de résoudre certaines difficultés liées à la propriété intellectuelle ou à d'autres dispositions légales (comme

l'exportation de certains produits) ;

- une procédure d'audit de codes. Pour ce faire, il peut être très utile d'utiliser les outils réalisés pour faciliter ces missions : notamment le logiciel open source Fossology qui peut permettre un examen systématique de code et l'outil commercial Protex de la société Black Duck qui offrira une analyse beaucoup plus poussée³⁵. Les deux solutions³⁶ peuvent utilement être combinées : la première étant systématique et la seconde préalable à certaines mises à dispositions ou distributions du code ;
- la labellisation de composants et de licences. Cette politique de labellisation permettrait d'accélérer la réutilisation de composants, modifiés ou non, déjà audités et considérés comme conformes à la politique du groupement (techniquement et juridiquement), alors que le suivi d'une procédure particulière obligatoire accompagnerait l'utilisation de composants non encore certifiés. La même pratique pourrait être élaborée au profit des licences libres et open source a priori considérées comme « conformes à la politique du projet/du groupement ». Une procédure particulière serait alors attachée à l'acceptation de composants open source soumis à d'autres licences (de nombreuses sociétés ont ainsi constitué en leur sein de tels catalogues de composants open source, éventuellement aidées pour cela par des logiciels comme l'outil Code Center de la société Black Duck).

2.1.1.3 Organiser l'accompagnement

À ce stade, il s'agit de définir quelles sont les tâches qui peuvent être externalisées et quelles sont celles qui doivent être conservées en interne.

2.1.1.3.1 La création d'une structure ad hoc

Une externalisation totale semble difficile en raison de l'interpénétration des notions juridiques et techniques d'une part, et de la criticité de ces enjeux d'autre part. Tout au plus est-il possible de penser à l'intervention de tiers quant à l'audit initial (en cas de solution préexistante), à l'édiction d'une gouvernance, voire en préalable à toute distribution — pour le reste, une personne, voire une équipe interne, doit être référente afin de résoudre (ou transmettre à cette fin) les difficultés qui se présentent à elle et « infuser » les bonnes pratiques. Une solution intermédiaire pourrait être de rejoindre une structure plus grande (comme une fondation) qui permette de mutualiser une partie de la fonction.

L'internalisation elle-même peut prendre différentes formes, qu'il s'agisse de créer une entité ad hoc (référent open source, comité open source, etc.) ou de modifier les structures existantes pour y intégrer ces besoins, notamment par l'intégration de règles générales, comme la définition de politique en matière de licences, la validation de paquets logiciels, etc.. De nombreux paramètres

35 La société éditrice propose un certain nombre d'offres pour les projets communautaires — à commencer d'ailleurs par l'Eclipse Foundation.

36 La liste n'est néanmoins pas exhaustive et d'autres produits similaires existent, édités par des sociétés américaines (telles Palamida, OpenLogic ou ProteCode) ou françaises (comme Antelink).

entrent dans cette équation, mais il est généralement conseillé, lorsque la taille le permet, de commencer par la mise en place d'entités autonomes, de sorte qu'elles s'éclipsent une fois la nouvelle culture infusée et les réflexes intégrés — au risque sinon de buter sur la résistance des collaborateurs (surtout dans le cas du passage d'une politique de licence traditionnelle à une politique ouverte : la longue domination du paradigme antérieur, engendrant de nombreux réflexes et habitudes profondément ancrés, peut entraîner le rejet irrationnel de ce mode alternatif d'édition).

Quelle que soit l'organisation choisie, il est nécessaire que cet accompagnement s'appuie sur des relais locaux (scouts), des correspondants aptes à faire remonter et redescendre les informations utiles.

À la fois structure de référence et structure inter-services, un tel organisme permet de mutualiser le travail d'accompagnement, de rationaliser les décisions, de définir des politiques globales et d'automatiser le traitement des questions. Par ailleurs, pour qu'une telle structure puisse être suivie d'effet, il est nécessaire qu'elle soit facilement joignable, qu'elle bénéficie d'un pouvoir décisionnel et, dans l'idéal, qu'elle soit suivie par un membre de la direction. Enfin, elle seule doit être consultée pour autoriser et décider de l'utilisation de briques logicielles sous licences libres / open source et de la diffusion sous de telles licences (quitte à ce que ses décisions soient avalisées par la direction).

2.1.1.3.2 Gérer les contributions au sein de la société

Comme nous l'avons vu, les licences s'étendent graduellement à l'ensemble des droits de propriété intellectuelle. Une vigilance accrue s'observe donc chez les sociétés qui mixent une culture libre (open source) et propriétaire : l'enjeu est alors de délimiter très précisément (généralement par des politiques open source très soignées), voire de cloisonner, les portefeuilles de droits relatifs au projet afin de s'assurer que l'activité open source ne vienne pas impacter et dévaloriser certains droits de propriété intellectuelle qui restent très attractifs pour la société.

C'est ce type d'attention qui explique par exemple la renonciation au bénéfice de la dévolution légale de droits relative à certains développements qui doivent être reversés à des projets placés sous des licences trop contraignantes (au profit des salariés qui conservent ainsi leurs droits)³⁷.

De la même façon, une responsabilité particulière pèse sur les services juridiques qui doivent s'assurer que les engagements contractuels avec les sous-traitants, fournisseurs, etc. soient bien respectés et maintenus lors de la diffusion sous licence libre et open source.

Pour une société, il arrive parfois qu'il soit plus simple, voire nécessaire, d'envisager une cession de droits au profit de projets qui assureront le maintien et l'intégration des différentes contributions.

Dans cette solution, tant l'entreprise que le salarié seront généralement signataires d'un document

³⁷ On peut ici citer l'exemple d'HP vis-à-vis de la GNU GPL v. 3 (qui pourrait impacter la valeur de son portefeuille de brevets).

de cession de droits au profit du projet. Il peut aussi être envisagé, lorsque la mission du salarié l'exige ou en raison du contrat particulier qui le lie à la société, que celle-ci lui délègue le pouvoir de mettre sous licences libres les contributions réalisées dans le cadre de ses missions salariales : le risque est faible pour l'entreprise lorsque celui-ci travaille dans un tel environnement, et cela lui permettra de sécuriser les contributions de son salarié vis-à-vis des communautés et des projets dans lesquels il intervient.

2.1.1.3.3 Faire travailler ensemble les différentes équipes

De façon pragmatique, lorsque l'on exploite du contenu sous licence libre en entreprise ou ailleurs, il est nécessaire d'adapter la procédure de validation juridique et contractuelle, l'analyse devant être réalisée autant en amont des projets qu'en aval. Un travail est nécessaire dès en amont afin de déterminer la faisabilité juridique des propositions techniques et s'assurer que les engagements puissent être respectés (en gardant en tête le leitmotiv : « on ne peut donner plus de droits que l'on en possède », c'est d'ailleurs cette seule idée qui ressort des contentieux français ayant mis en jeu la GNU GPL). Enfin, une dernière étude juridique est nécessaire en aval pour confirmer le choix final de la licence et procéder, s'il y a lieu, aux ajustements contractuels en fonction du produit, des attentes et bien évidemment des licences en présence.

Ainsi, ce travail ne peut être mené que conjointement entre l'équipe technique, seule apte à fournir les informations détaillées sur le logiciel, et l'équipe juridique. Les données brutes nécessaires à l'étude juridique sont : le détail des briques logicielles utilisées, leur(s) licence(s) respective(s) (comprenant leur version, et toute autre information influant sur ces licences : restriction, interprétation, exception, etc.), et enfin les interactions existantes entre chacune de ces briques (en pratique, un schéma). Un dialogue s'établit par la suite pour affiner les derniers détails, et de nombreuses navettes permettront de compléter l'analyse.

D'une manière plus générale, le succès d'une politique active en faveur de l'open source repose sur une implication de l'ensemble des collaborateurs dans le processus de restructuration et une montée des compétences de ces derniers afin qu'ils comprennent et partagent ce nouveau paradigme : par l'organisation de formations, la rédaction et la mise à disposition des informations qui leurs sont utiles, l'organisation de leurs contributions au sein de chartes (contribution, développements, processus de validation, etc.).

2.1.1.3.4 S'équiper avec de bons outils

Enfin, l'usage de certaines solutions logicielles peut simplifier, voire approfondir³⁸ cette mission, en automatisant l'audit de code³⁹ voire la gestion au quotidien de la base de code validée. Tous les outils ne sont pas équivalents, et tous n'ont pas vocation à être utilisés au même moment ou par les mêmes personnes (dans le cadre de due diligence, les audits demanderont ponctuellement l'utilisation d'outils très poussés, alors que la vérification automatique et journalière des

³⁸ Par exemple avec une comparaison du code du projet avec celui de leur base.

³⁹ Voir à cet égard les solutions décrites dans le Guide Open Source, partie 1.2.2. « Description du code source », op. cit..

développements réalisés pourra être plus superficielle).

Enfin, le projet Software Package Data Exchange⁴⁰ (SPDX⁴¹) a récemment introduit un format standard pour communiquer les éléments, les licences et droits d'auteur associés à un logiciel (et ainsi faciliter la prise de connaissance des obligations et le respect des licences). Ainsi, chaque société ayant réalisé ce travail d'audit pourra dorénavant accompagner la livraison de son logiciel d'un fichier SPDX qui recensera tous les composants utilisés et leur(s) licence(s).

2.1.2 Gérer les autres aspects juridiques

Comme vu précédemment, différents documents peuvent être mis en place pour accompagner l'organisation vis-à-vis des tiers et notamment pour donner un cadre formel à la collaboration (1.2.2.1), et la rédaction d'une politique en matière de marque précise à l'intention des tiers et des licenciés (1.2.2.2).

2.1.2.1 Structurer le projet

Que le projet soit ouvert ou fermé, la mise en place d'un cadre propice à son développement et son exploitation nécessite une réflexion quant à l'organisation du groupement autour du projet et vis-à-vis des tiers.

Tant que l'organisation du projet est suffisamment attractive pour motiver des contributions extérieures et qu'il répond aux contraintes des mainteneurs, il n'y a aucune raison de changer celle-ci.

Si le projet souhaite s'ouvrir de manière à impliquer plus de contributeurs, plusieurs schémas peuvent être construits pour organiser les développements collaboratifs, avec leurs avantages et inconvénients (création d'une entité juridique distincte, mise en place d'un consortium, l'appui d'un partenaire leader — voire une absence délibérée). Notons que la mise en place d'un consortium est la solution la plus légère (généralement la première utilisée par les sociétés qui cherchent à collaborer en raison de la faiblesse de l'engagement), éventuel premier jalon d'une autonomisation quasi-systématique des projets dès lors qu'ils gagnent en contributeurs.

Dans l'hypothèse de la libération d'un projet, il est préférable de structurer un minimum dès le début (donner simplement l'accès au code et le processus pour contribuer) quitte à mettre ensuite en place une organisation adaptée.

Enfin, la gouvernance des groupements est généralement découpée entre les aspects stratégiques, les aspects techniques et les aspects communication/sensibilisation (lien avec les utilisateurs et la communauté).

40 Le projet SPDX (pour Software Package Data Exchange) est un format standard pour la communication des composants, licences et mentions légales associées à la distribution d'un logiciel. Le site a été réalisé sous la direction de la Linux Foundation, avec la participation de nombreuses sociétés parmi lesquelles Alcatel-Lucent, Antelink, Black Duck Software, Canonical, HP, Micro Focus, Motorola Mobility, nexB Inc, OpenLogic, Palamida, Protecode, Source Auditor, Texas Instruments et Wind River.

41 <http://spdx.org>

2.1.2.2 Les politiques en matière de marques

Le droit des marques retrouve toute sa noblesse dans l'industrie de l'open source, puisqu'il reste bien souvent le meilleur garde-fou pour empêcher certains types d'exploitation d'un projet qui est pourtant librement disponible (notamment en termes d'image, mais aussi de comportement commercial). Néanmoins, il est parfois nécessaire de relativiser l'usage qui peut en être fait au regard des prérogatives qu'il confère à son titulaire, car la tentation est grande pour certains d'essayer de récupérer là un contrôle qu'ils ont abandonné en faisant le choix de l'open source.

Ainsi, les politiques en matière de marque sont généralement des documents publics qui précisent quelles sont les marques détenues par le projet et les conditions de leur utilisation : une seule marque pouvant être utilisée différemment selon le contexte (lorsque la création est modifiée ou non) ou plusieurs marques pouvant être utilisées pour différencier les usages permis sur chacune (par exemple en autorisant seulement les partenaires à utiliser la marque figurative).

Les meilleurs exemples existants, à cet égard, sont ceux des projets du type OpenOffice.org / LibreOffice, Firefox, GNU/Linux, etc., qui réunissent à la fois entreprises, associations et fondations, contributeurs individuels et le grand public.

2.1.2.3 Politique en matière de brevets

La manipulation des brevets est généralement très critiquée dans l'industrie de l'Open Source. Ainsi, en plus de clauses spécifiques ajoutées aux licences libres et Open Source, une pratique défensive s'est développée.

En réaction au danger que constituent les patent trolls, on observe des actions de regroupements de brevets à titre défensif — initiatives individuelles⁴² ou collectives⁴³ — avec pour objectif d'immobiliser toute action tierce en contrefaçon de brevets logiciels. Allant généralement encore plus loin, les initiatives peuvent associer une promesse de ne pas opposer leurs brevets aux communautés open source et l'engagement d'une protection des communautés grâce à ces portefeuilles de brevets — avec pour objectif de résoudre l'insécurité juridique qui découle des brevets logiciels. L'[Open Invention Network](#) a conduit une initiative intitulée Linux Defenders : consortium composé d'industries de haute technologie (notamment IBM), le projet a pour vocation d'aider à protéger les communautés de logiciels open source contre les dangers et attaques récurrentes en matière de brevets. Cette action est destinée à réduire la crainte des actions injustifiées (sur le fondement de titres très faibles susceptibles d'être annulés par un juge), mais excessivement coûteuses pour les individus, les communautés et les petites ou moyennes entreprises

42 Plusieurs milliers de brevets sont ainsi regroupés en garantie par des sociétés comme IBM, Nokia, Sun, etc. — notamment l'IBM Statement of Non-Assertion of Named Patents Against OSS, le Sun patent program ou encore le Novell Statement on Patents and open source Software.

43 Notamment l'Open Source Development Laboratory (OSDL), l'Electronic Frontier Foundation (EFF — avec notamment l'initiative Patent busting project) ou l'Open Invention Network (OIN) auquel ont souscrit des sociétés comme Sony, IBM, NEC, Red Hat, Philips et Novell. La Foundation for a Free Information Infrastructure (FFII) et l'initiative « no software patents » affirmant que la constitution de regroupements de brevets à titre défensif est une mauvaise solution par rapport à la lutte pour imposer la

(au point de les faire plier sous le seul prix du procès). Il consiste en la réalisation d'une publication défensive (avec des appels à la communauté), la transmission de ces données aux organismes nationaux de propriété intellectuelle afin qu'ils puissent avoir connaissance d'une antériorité susceptible d'invalider un brevet.

Le rôle et l'action de l'OIN ne cessent de croître et, en 2011, le Département de Justice américain a ainsi consacré son rôle lors du rachat de Novell par un consortium au sein duquel se trouvait notamment Microsoft. Il était question de plus de 880 brevets. Le Département de la Justice a soumis ce rachat à la condition que l'ensemble des 882 brevets soit reversé à l'Open Invention Network, brevets qui viennent compléter le portefeuille conséquent de l'OIN.

Licence

Ce livre blanc a été écrit par Benjamin Jean pour la société Inno³ sous licence Creative Common By SA 4.0. Pour obtenir des informations complémentaires sur cette licence, veuillez consulter le site : <http://creativecommons.fr/>

À propos d'inno³



Inno³ est un cabinet français indépendant accompagnant les organisations souhaitant tirer pleinement profit de l'Open Source/ au sein de leur IT et de leur R&D.

Leader du domaine, inno³ dispose de toutes les compétences nécessaires pour adresser les trois piliers sur lesquels repose cette mutation : la dimension outil (par la gestion des progiciels Open Source), la dimension juridique (au travers de la gestion des licences Open Source et des contrats associés) et la dimension humaine (dans l'organisation des rapports avec les communautés dont sont issus les projets). Les offres de services proposées vont de la simple expertise ponctuelle (notamment audit et analyse d'opportunité) à l'accompagnement au changement (définition de stratégie, mise en place de politique et gouvernance dédiées) et au transfert de compétences (au travers de cursus préconstitués ou de modules « à la carte »).

La société anime enfin un réseau d'experts métiers qui permettent de porter les transformations liées à l'Open Source dans tous les secteurs industriels ou publics.

À propos de l'Auteur



Fondateur du cabinet inno³, Benjamin Jean travaille essentiellement sur les nouveaux usages et collaborations qui se développent autour de la création et de l'innovation (Open Source, Open Data ou plus généralement toute la mouvance en faveur d'une innovation ouverte). Actif depuis près de dix ans dans ce domaine, il a notamment œuvré pour leur prise en compte au sein des gouvernances de multiples acteurs privés et publics, de tailles et de domaines variés, et formé de nombreuses équipes internes aux aspects juridiques afférents.

Il enseigne par ailleurs la propriété intellectuelle au sein de plusieurs Masters spécialisés, intervient comme consultant au sein du cabinet Gilles Vercken et maître de conférences à Sciences Po.

Il est administrateur de PLOSS Paris Région, membre du Directoire de Systematic et Président de l'association *Open Law, le droit ouvert*. Il est également très présent dans les communautés du Logiciel Libre, administrateur de Framasoft et cofondateur de Veni Vidi Libri.