

Le droit des licences Open Source

Quelles règles pour les entreprises ?

Sous la direction de Me Bernard LAMON, avocat spécialiste
en droit informatique et télécommunication.

Version 1.1, août 2009.

*Vous pouvez citer des courts extraits de ce livre blanc en citant
l'auteur et son emplacement sur www.bernardlamon.fr.*

août 2009

Chapitre 1 PANORAMA DES LICENCES LIBRES.....13

1.1 L'environnement juridique de la licence libre	13
1.1.1 Lien entre l'acquisition de la qualité d'auteur et la licence libre.....	14
1.1.2 La soumission à une licence libre.....	15
1.1.3 Hypothèse d'une contradiction entre une licence libre et les dispositions légales impératives.....	17
1.2 Le dénominateur commun des licences libres.....	19
1.3 La licence GPL	21
1.3.1 Forces de la licence GPL	21
1.3.2 Faiblesses de la GPL : le nécessaire avènement d'une version 3 de la licence	22
1.3.3 Menaces.....	24
1.3.4 Opportunités.....	25
1.4 Les autres licences	27
1.4.1 La famille BSD	27
1.4.2 Licences spécifiques.....	29
1.5 La question de la compatibilité entre différentes licences.....	31
1.5.1 La compatibilité entre licences GNU	32
1.5.2 La compatibilité entre licences de la famille BSD	33
1.5.3 La compatibilité entre licences GNU / BSD	33

Chapitre 2 LE DROIT FRANCAIS.....35

2.1 La paternité du logiciel libre.....	35
2.1.1 Le logiciel, une œuvre de l'esprit protégée.....	35
2.1.2 La protection de la paternité en droit commun et en matière de logiciel.	36
2.1.3 La paternité du logiciel libre dépend des modalités de son développement.	37
2.2 L'œuvre dérivée et le logiciel libre.	40

Chapitre 3 LICENCE GPL42

3.1 Les principaux aspects de cette licence sont que :.....	43
3.2 Distribution et GPL	44
3.2.1 Distribution et codes sources	44
3.2.2 Distribution et transmission de licence	47
3.3 Sur l'aspect viral de la GPL.....	48
3.4 GPL et œuvre dérivée	49
3.5 Sur la validité française de la GPL	50

Chapitre 4 La jurisprudence.....52

4.1. Jurisprudence américaine.....	52
4.1.1 District Court of Massachusetts, Progress Software Corp. v My SQL AB, 28 février 2002	52
4.1.2 Wallace v. International Business Machines Corp. et al., 20 mars 2006 et 16 mai 2006	53
4.1.3 Jacobsen v. Katzer, 13 août 2008.....	54
4.2. Jurisprudence allemande.....	54
4.2.1 District Court of Munich I, 19 nov 2004.....	54
4.2.2 District Court of Munich, Welte vs. Skype Technologies SA, juin 2007.....	55
4.3. Jurisprudence française.....	55
4.3.1 TGI Paris, 28 mars 2007, Educaffix c/ CNRS, Université Joseph Fournier et autres	55
4.3.2 TGI Chambéry, Société ERN c/ Conseil Général de Savoie et Université de Savoie, 15 nov. 2007.....	58
4.3.3 Assignation de la S.A.S. Free devant le Tribunal de Grande Instance de Paris, novembre 2008.....	61

INTRODUCTION

L'objectif du présent livre blanc est de vous offrir un outil opérationnel de découverte des enjeux juridiques des logiciels et œuvres « open source ». Il s'adresse prioritairement aux :

- directeurs des systèmes d'information, responsables informatiques et chefs de projet, et responsables du développement,
- que ce soit au sein d'entreprises éditeurs de logiciels
- ou d'entreprises utilisatrices de logiciels
- ou de collectivités publiques utilisatrices de logiciels.

Dans son parcours quotidien, le lecteur croise régulièrement des logiciels dont on lui indique qu'ils sont « libres » ou « open source ». Dans cette introduction, nous verrons que le terme de « libre » est susceptible d'induire en erreur.

Dans tout ce livre blanc, nous utilisons indifféremment l'expression : « logiciel libre » ou « open source ».

Le droit d'auteur.

Le droit d'auteur partout dans le monde repose sur quelques principes essentiels : l'auteur d'une œuvre de l'esprit bénéficie d'un droit de propriété sur l'œuvre par la loi. Deux grands systèmes coexistent : le droit d'auteur, né en France au XVIIIème siècle, et le droit du copyright, né juste avant en Grande-Bretagne.

Les termes, les mots ont énormément d'importance (parfois trop ?) pour les juristes.

Le droit d'auteur est axé sur la personne de l'auteur : l'individu est particulièrement protégé (notamment contre le producteur, considéré comme une grande entreprise coupable par avance de tous les maux). Il est doté non seulement de droits patrimoniaux (qui s'expriment en espèces sonnantes et trébuchantes) mais aussi de droits moraux (le droit au respect de son œuvre, à ce que son nom soit toujours cité, le droit de publier l'œuvre et de revenir sur sa décision). Même quand une œuvre est créée en groupe (au sein d'une entreprise, par exemple), la personne morale (société ou association) n'est que très rarement dotée des droits d'auteur.

Quant au copyright, c'est non pas le droit des auteurs, mais le droit des imprimeurs à l'origine (copy right = droit de copie). Ce droit est orienté principalement vers l'exploitation commerciale de l'œuvre. L'auteur a surtout des droits patrimoniaux, quasiment pas de droits moraux. Il peut les vendre (quitte à ce que le prix soit ridiculement bas...).

Cette présentation (à grands traits) convient bien pour les œuvres classiques (pièces de théâtre, livres, chansons, et même les films), mais il a fallu procéder à quelques adaptations dans les années 1970 lorsqu'on a décidé de protéger le logiciel par le droit d'auteur ou le copyright. Ainsi, en France, les droits moraux sont réduits à peu de chose. La seule chose que l'auteur d'un logiciel puisse exiger est qu'il « signe » son œuvre. En pratique, c'est fréquemment réalisé par une insertion dans les codes source sous forme de « remarques » (partie de code source

non compilable et qui permet de suivre, par la lecture des commentaires, la construction du logiciel).

Quand le logiciel libre est apparu en France dans les années 1990 (son expansion a accompagné celle de l'internet, les deux se nourrissant réciproquement), les aspects juridiques étaient totalement négligés.

D'ailleurs, de manière générale, il n'était pas attendu d'un DSI ou d'un développeur informatique un quelconque intérêt à la chose juridique. De ce côté-là, les choses ont évolué depuis 15 ans. Les explications sont multiples : maturité des acteurs, augmentation de l'offre en droit de l'informatique, quelques contentieux fracassants, une sensibilité plus générale à la chose juridique dans toute la société.

Il y a encore cinq ans, lorsque j'évoquais cette question dans mes conférences (j'en assure en moyenne une cinquantaine par an, de la petite présentation de 20 minutes à la conférence complète sur une journée), l'intérêt était poli... ou inexistant.

Les choses ont commencé à changer après l'explosion de la bulle internet aux Etats-Unis. Le prix de certaines sociétés vendues a été sérieusement revu à la baisse parce que des logiciels devaient être re-développés pour éviter l'effet viral de la GPL et conserver la qualité de logiciel propriétaire (*"When IBM acquired Think Dynamics, a painstaking manual examination of its code revealed 80 to 100 examples of open source code that Think Dynamics programmers had passed off as their own. As a result, the price of that company went down from 67 million dollars to 46 million--not a happy moment for its owners and shareholders, I'm sure."*)

(www.oreillynet.com/pub/wlg/4291#infringers).

Si ce livre est entre vos mains, c'est que ce mouvement vous concerne.

Le logiciel et la licence.

Pour la clarté du propos de ce livre blanc, il faut rappeler qu'un logiciel est une œuvre de l'esprit, qui peut se définir comme une suite d'instructions qui s'expriment en langage compréhensible par une machine (code objet) ou par un être humain (code source).

Le droit d'auteur sur un logiciel appartient à son auteur (article L 111-1 du code de la propriété intellectuelle) qui peut être une personne physique (un individu) ou morale (une société commerciale, par exemple).

Le contrat par lequel l'auteur du logiciel permet à un utilisateur d'exécuter un logiciel est une licence (en droit US, la question de savoir si une licence est un contrat est débattu, mais l'opinion majoritaire va dans ce sens). La licence prévoit précisément les conditions d'utilisation du logiciel : exécution, modification du code, accès au code source...

Quelques mots d'histoire...

Selon la légende, le logiciel libre serait né parce qu'un utilisateur mécontent (Richard Stallmann) ne parvenait pas à obtenir le code source d'un logiciel de pilote d'imprimante. Les choses sont peut-être un peu plus complexes. A l'origine, les logiciels étaient presque tous sans protection, et cela ne gênait personne parce que la

valeur de l'informatique était dans le matériel (hard) pas dans le logiciel (soft).

A mesure que la chaîne de valeur s'est déplacée du hard au soft, les logiciels sont devenus de plus en plus « propriétaires ». Plus précisément, les propriétaires des logiciels ont commencé à se préoccuper de leur protection, et ils ont décidé, notamment, de ne plus livrer les codes source avec leurs logiciels exécutables.

Une des phases marquantes de l'histoire du logiciel est la création de la FSF (free software foundation) en 1985, et la publication de la GPL-GNU v2 quelques années plus tard.

Quelques incompréhensions.

Tout d'abord, certains pensent qu'un logiciel libre est nécessairement un logiciel gratuit. Le risque est encore plus grand si on pense à l'expression anglaise (« free » signifie libre et aussi gratuit). Cette incompréhension est levée par une célèbre expression : « free as in free speech, not as in free beer », ce que l'on pourrait traduire par : le logiciel est libre dans le sens liberté d'expression, pas gratuité.

Les licences libres : pas seulement pour du logiciel.

Depuis quelques années, un mouvement similaire à celui du logiciel « open source » est né concernant les autres œuvres protégées par le droit d'auteur (les licences Creative Commons, par exemple).

Le casse-tête de l'aspect international du logiciel libre.

Par nature, depuis l'émergence d'Internet, le logiciel libre est un logiciel développé et distribué par Internet, donc sans aucun souci des frontières nationales. Il n'est pas rare que sur un projet « open source » collaborent des contributeurs de nombreuses nationalités différentes.

Or, dans ce cas, avant de déterminer les conséquences de la loi (ai-je le droit de faire ceci ou cela, comment me protéger etc...), il faut déterminer la loi nationale applicable. Même si le droit d'auteur est régi par des principes qui sont identiques partout dans le monde, les détails d'application varient de pays à pays.

Or, il est déjà difficile de se prononcer dans des situations simples : œuvre composée en Belgique par un auteur français et exploitée en France. Cela devient parfois très difficile si les co-auteurs (contributeurs au logiciel) habitent dans de nombreux pays différents.

Cette complexité ne doit pas, en pratique, être surestimée. Comme les décisions de justice qui ont été rendues jusqu'à aujourd'hui le démontrent, ces questions ne sont pas souvent centrales.

Quel risque ?

Depuis que je parle de logiciel libre dans mes conférences, et au cours de rendez-vous avec des clients, la question qui revient sans cesse est celle du risque réel encouru.

Tant qu'aucune décision de justice n'était rendue, on en restait aux évaluations et aux conjectures. Les premières

décisions ont conforté mes hypothèses. Le risque peut venir de trois sources au moins.

Les « hippies ».

Sous ce vocable, je classe de manière humoristique (et non pas péjorative) les actions en justice menées par les partisans du logiciel libre. Ainsi, les décisions obtenues par M. Harald Welte en Allemagne et l'action menée en France contre FREE.

L'instrumentalisation et le coup de boomerang !

Dans un conflit entre sociétés commerciales, l'une reproche à l'autre une contrefaçon de logiciels. Pour éviter une condamnation (parce qu'elle a copié les logiciels de la première), l'accusée expose que le logiciel invoqué par la première est composé de briques de logiciel libre. Dans un tel cas, le procès tombe.

Dans une affaire que j'ai eu à traiter il y a déjà quelques années, le procès a ainsi été arrêté avant même d'avoir commencé.

Le discount.

Le dernier risque est celui que j'évoquais ci-dessus : la baisse du prix des actions d'une société éditeur de logiciel. Sauf erreur de ma part, l'audit préalable à l'acquisition d'une société éditeur ne comprend pas encore en France l'examen des codes source. C'est maintenant une pratique courante aux Etats-Unis. A bon entendeur...

Organisation du livre blanc.

Chaque chapitre est destiné à permettre une lecture rapide. Les éléments essentiels sont placés dans des passages encadrés.

Le premier chapitre dresse un panorama des différentes licences, et leurs forces et faiblesses respectives.

Puis on envisage la confrontation des licences libres avec deux notions centrales du droit français : le droit à la paternité de l'auteur du logiciel (le droit de se dire le « papa » ou la « maman » du logiciel) et la notion d'œuvre dérivée.

Les chapitres suivants reviennent sur les questions essentielles posées par la GPL sur le plan juridique : l'effet viral de la GPL, la notion d'œuvre dérivée et de distribution.

Enfin, pour rester très pratique, j'ai décidé de regrouper les différentes décisions rendues en la matière que ce soit en France, en Allemagne, ou aux Etats-Unis. Pour les rendre plus accessibles, nous vous en proposons un résumé, avec aussi un résumé de l'assignation qui a été délivrée contre Free devant le tribunal de grande instance de Paris.

J'espère que cette lecture vous permettra d'y voir plus clair sur cette question peu explorée qui risque de provoquer de véritables séismes dans le futur.

Je termine cette introduction en remerciant tous les contributeurs de ce livre blanc : tout d'abord, Eric LE QUELLENEC, avec qui je travaille depuis plus de cinq ans, et trois stagiaires qui ont accepté avec enthousiasme de

contribuer à ce livre : par ordre alphabétique, Kevin ARNAUD, Ronan DELCROIX, et Ludovic de la MONNERAYE.

Chaque chapitre a été préparé par l'un d'entre eux, j'ai relu et amendé tous les chapitres. Quant à l'introduction, je l'ai écrite et elle a été relue et amendée par les autres.

CHAPITRE 1

PANORAMA DES LICENCES LIBRES

Une licence, qualifiée ou non de libre, n'est ni plus ni moins qu'un contrat conclu entre un développeur informatique ayant qualité d'auteur et un utilisateur final et/ou un autre développeur. C'est pourquoi la licence libre, comme n'importe quel autre contrat, s'inscrit dans un environnement juridique donné et doit respecter (en principe) certaines contraintes légales (1.1). Sous ces réserves légales, chaque auteur d'un logiciel peut choisir voire composer sa propre licence. Cependant, pour être qualifiée de libre, chaque licence doit répondre à des critères communs mis en exergue par l'OPEN SOURCE INITIATIVE (1.2). Le plus souvent, l'auteur d'un logiciel qu'il souhaite « *open source* » choisira une des licences préexistantes, telles que la GPL (1.3) ou l'une des autres licences disponibles (1.4). Que l'on soit développeur ou utilisateur, chaque licence disponible est potentiellement porteuse de forces, faiblesses, menaces ou opportunités. Dans tous les cas, sur le terrain, la question de la comptabilité entre différentes licences peut être très délicate (1.5).

1.1 L'environnement juridique de la licence libre

Contrairement à de nombreuses idées reçues, les licences libres ne se sont pas construites en rupture avec l'environnement juridique pré-existant.

En effet, pour qu'il y ait licence, il faut qu'il y ait reconnaissance d'un droit : pour le logiciel il s'agit du droit d'auteur (ou « *copyright* » dans les pays anglo-saxons de tradition de « *common law* »), parfois du brevet en droit américain ou dans certains cas plus ou moins bien déterminés en Europe.

En revanche, dans la querelle entre protection du logiciel par le brevet ou le droit d'auteur, les partisans du libre ont clairement choisi leur camp. La protection par brevet est clairement rejetée en ce qu'elle enferme le code source breveté à la fois dans un environnement technique donné et pour des applications figées dans le brevet.

La souplesse du droit d'auteur s'accorde beaucoup mieux à la conception « libérale » des licences libres. Après un bref rappel des conditions d'acquisition de la qualité d'auteur (1.1.1), il convient d'analyser comment un auteur peut se soumettre à une licence libre (1.1.2). La gestion d'une éventuelle contradiction entre une disposition légale impérative et une stipulation d'une licence libre doit aussi être évoquée (1.1.3).

1.1.1 Lien entre l'acquisition de la qualité d'auteur et la licence libre

Aux termes de l'article L.111-1 du code de la propriété intellectuelle, l'auteur d'une œuvre est, par présomption, celui sous le nom duquel l'œuvre est diffusée : en principe, nul besoin de faire la moindre démarche pour voir son logiciel protégé tant en France que dans le monde entier (ou du moins tous les Etats membres de l'Organisation Mondiale du Commerce (OMC) ou ayant ratifié la convention de Berne du 9 septembre 1886).

Comme pour un logiciel propriétaire, l'auteur d'un logiciel libre peut être une personne physique, une association à but non lucratif, une société en tant qu'employeur de salariés dont les droits sont automatiquement transmis sur le fondement de l'article L.113-9 du code de la propriété intellectuelle (ci-après CPI).

De plus, le logiciel libre peut être une œuvre de collaboration ou de coopération. Dans le premier cas, chaque contribution de chaque auteur est clairement identifiable. Dans le second cas, les contributions des différents coauteurs se confondent.

Il ne faut pas oublier que la qualité d'auteur est tout de même conditionnée par le respect de certaines qualités intrinsèques à l'œuvre, à savoir la condition d'originalité. En matière de logiciels, cette condition est toutefois appréciée de manière peu exigeante : il faut caractériser la marque de l'empreinte de la personnalité de l'auteur dans le logiciel développé. Plus précisément, pour le logiciel, le code développé ne doit pas découler d'une logique contraignante mais refléter "l'apport intellectuel" de son auteur (Cass. ass. plén., 7 mars 1986 : Juris-Data n° 1986-098125).

L'auteur d'un logiciel qui entend le soumettre à une licence libre doit donc préalablement s'assurer qu'il répond à ces conditions.

1.1.2 La soumission à une licence libre

C'est parce qu'il est auteur au sens juridique du terme qu'un développeur peut décider de placer ou non son logiciel sous licence libre. Toutefois, la grande liberté reconnue à l'auteur ne s'applique pas de la même façon pour les droits patrimoniaux ou pour les droits moraux.

Les droits moraux

En application de l'article L.121-1 alinéa 3 du CPI, les droits moraux sont perpétuels, inaliénables et imprescriptibles. Licence libre ou pas, un auteur ne peut donc se départir de ses droits moraux facilement.

Traditionnellement, les droits moraux sont au nombre de cinq, en droit des logiciels seuls trois prérogatives sont applicables :

- droit de divulgation : porter ou pas l'œuvre à la connaissance du public ;
- droit à la paternité : mentionner le nom de l'auteur sur l'œuvre et faire respecter cette mention par les tiers ;
- droit au respect de l'œuvre et à son intégrité : ce droit ne peut s'appliquer pour un logiciel en raison des caractères évolutif et interfaçable du code source (article L.122-7 du CPI) ;
- droit au retrait et de repentir : l'article 121-7 2° du CPI dispose que ces droits moraux ne s'appliquent pas pour le logiciel.

En France, un développeur qui souhaite renoncer à ses droits moraux peut le stipuler par contrat, la clause sera nulle de plein droit et il peut revenir sur sa renonciation sans porter l'affaire devant les tribunaux.

La plupart des licences libres disponibles ne sont pas en rupture avec ces droits moraux. Bien au contraire, le droit à la paternité d'un logiciel est au cœur de la GPL, alors qu'en droit du logiciel américain, le droit moral est pratiquement inexistant.

Droits moraux et licences libres font plutôt bon ménage. Le choix du passage d'un logiciel sous cette licence libre impacte directement la mise en œuvre et la gestion des droits « patrimoniaux ».

Les droits patrimoniaux

L'article L.122-6 du code de la propriété intellectuelle permet à l'auteur d'effectuer et d'autoriser :

« 1° La reproduction permanente ou provisoire d'un logiciel en tout ou partie par tout moyen et sous toute forme. Dans la mesure où le chargement, l'affichage, l'exécution, la transmission ou le stockage de ce logiciel nécessitent une reproduction, ces actes ne sont possibles qu'avec l'autorisation de l'auteur ;

2° *La traduction, l'adaptation, l'arrangement ou toute autre modification d'un logiciel et la reproduction du logiciel en résultant ;*

3° *La mise sur le marché à titre onéreux ou gratuit, y compris la location, du ou des exemplaires d'un logiciel par tout procédé. »*

La licence libre n'est qu'une manière de déterminer comment ces droits patrimoniaux sont organisés.

1.1.3 Hypothèse d'une contradiction entre une licence libre et les dispositions légales impératives

En doctrine, il a été un temps discuté de la licéité des dispositions de certaines licences libres (M. Christophe CARON, DALLOZ 2003, P. 1556).

Le fait que l'utilisateur licencié ne manifeste pas expressément son consentement a été discuté. Le principe de l'acceptation tacite d'une licence n'est pas un réel

obstacle à l'opposabilité de la licence à l'égard de l'utilisateur.

Par ailleurs, il est exposé infra les nombreux débats qui ont eu cours sur le caractère viral de la licence GPL.

Les premières décisions allemandes, suivies des jugements français ont levé quelques unes des incertitudes juridiques sur l'éventuelle incompatibilité de certaines mesures des licences libres avec les dispositions impératives légales.

Toutefois, lors d'un procès, un juge n'a pas à faire l'audit détaillé d'un contrat de licence : il ne doit répondre qu'aux interrogations des parties et aux moyens développés par elles. Il est possible de citer deux exemples de questions en suspens :

- l'exclusion de toute garantie prévue dans la plupart des licences libres, peut être sérieusement mise en question à l'égard des consommateurs ;
- le fait que la plupart des licences libres soient rédigées en anglais pose une réelle question au regard de leur validité vis-à-vis des administrations publiques utilisatrices ou des consommateurs. En effet, la loi dite TOUBON du 4 août 1994 (n°94-665) impose l'usage du français à l'égard de ces deux catégories de cocontractants, sous peine de sanctions pénales voire de nullité.

En outre, les lois évoluent et ce qui était autorisé peut devenir prohibé.

C'est ainsi que l'adoption de la loi n°2006-961 du 1^{er} août 2006 relative au droit d'auteur et aux droits voisins dans la société de l'information (dite DADVSI) avait pu provoquer un grand émoi dans la communauté du logiciel libre.

L'introduction par cette loi du délit d'atteinte à une mesure technique de protection d'un droit d'auteur (mieux connue sous le nom de « DRM ») a été perçue comme une possible atteinte aux fondements mêmes des licences libres que sont l'accès aux codes sources et le droit de les modifier. En d'autres termes, programmer sous licence libre un DRM deviendrait un moyen de rendre d'empêcher toute réutilisation des codes sources.

Si dans l'exécution d'une licence, il devait exister une incompatibilité entre une disposition légale impérative et une disposition d'une licence libre, la solution mise en avant par la GPL est plutôt radicale (article 7 de la version 2 et dans des termes équivalents à l'article 12 de la version 3) :

« Si vous ne pouvez pas distribuer le programme en satisfaisant à la fois vos obligations liées à la licence et à d'autres obligations applicables, alors vous ne pouvez pas distribuer le programme du tout. »

Le principe d'un respect cumulatif des lois nationales et les termes de la licence libre est ainsi posé. En cas de contradiction frontale avec une disposition nationale. L'utilisateur peut utiliser le logiciel mais il ne peut pas le redistribuer.

1.2 Le dénominateur commun des licences libres

L'OPEN SOURCE INITIATIVE (OSI) a élaboré une définition en dix points de ce que doit être une licence libre (<http://www.opensource.org/docs/osd>). Pour être reconnue par le monde du libre, toute licence qui se prétend libre doit donc impérativement respecter ces critères.

Les trois premiers points de cette définition constituent réellement le dénominateur commun à toute licence libre. Une licence est libre si **la redistribution du logiciel est libre** : la licence ne doit pas interdire à qui que ce soit de vendre ou de transmettre le programme à des tiers.

La licence libre doit permettre **la distribution du logiciel sous forme de code source**. Si le code source n'est pas directement accessible, il doit être indiqué comment y avoir accès, moyennant, le cas échéant, remboursement des frais d'accès à ce code.

La licence doit permettre **des modifications et/ou des adaptations ou travaux dérivés du logiciel**. Ces travaux doivent être distribués sous les mêmes termes de licence.

Ces trois contraintes majeures peuvent paraître importantes. En réalité, il existe une marge d'appréciation. Ainsi, pour les travaux dérivés, la licence libre doit permettre de redistribuer les travaux sous la même licence : ce n'est pas forcément une obligation. C'est, par exemple, un critère permettant de distinguer la licence GPL d'une part, de la licence LGPL ou BSD d'autre part.

Les autres critères OSI posent des principes de bon sens : un logiciel libre ne doit pas être limité à un usage (d'où la contradiction avec la revendication d'un brevet logiciel évoqué supra), à une catégorie d'utilisateur (par exemple, les agences gouvernementales et pas les ONG ou inversement) ou à un type de technologie (APPLE vs PC, par exemple).

1.3 La licence GPL

Initié par Richard STALLMAN, très vite relayé par la FREE SOFTWARE FOUNDATION (FSF), le projet GNU a permis d'adopter plusieurs licences libres. Historiquement, la General Public Licence (ci-après GPL) a été la première licence adoptée en janvier 1989 et est désormais la plus répandue. Le terme GNU ne vise donc aucune licence précise, il faut toujours préciser la licence du projet GNU que l'on souhaite voir appliquer. Dans nos développements, nous utiliserons le terme de GPL pour désigner la licence GNU/GPL.

Que ce soit du point de vue utilisateur ou développeur, les forces faiblesses, menaces et opportunités (ci-après FFMO) de la licence GPL sont abordées successivement.

1.3.1 Forces de la licence GPL

Il s'agit de la licence la plus couramment employée. La plupart des statistiques disponibles évoquent le plus souvent un taux d'emploi de l'ordre de 70% pour l'ensemble des licences libres en cours.

La licence GPL est celle qui applique de la manière la plus complète l'ensemble des préceptes du mouvement libre : c'est ce qui explique son succès écrasant.

Ce qui a provoqué le succès de la GPL ainsi que sa « domination » actuelle tient en partie au mécanisme « viral » (ou « *share alike* ») qu'elle contient en son sein et qui implique qu'un logiciel encodé sous cette licence y soit lui-même soumis. Cette particularité quasiment inédite mérite qu'un chapitre complet du présent livre blanc lui soit consacré (voir infra).

1.3.2 Faiblesses de la GPL : le nécessaire avènement d'une version 3 de la licence

La licence GPL n'a pas été rédigée par des juristes mais d'abord et avant tout par des techniciens pour des techniciens.

C'est pourquoi, par rapport aux droits patrimoniaux de l'article L.122-6 du CPI précité, les notions présentes dans cette licence libre divergent quelque peu des définitions légales généralement employées.

Par exemple, en logiciel propriétaire, la notion de distribution concerne la commercialisation du logiciel. Dans la GPL, le terme de distribution est le fait de transmettre le logiciel, code source ouvert. Cette différence terminologique est une des causes de la confusion sur le caractère gratuit des logiciels libres.

Même si, dès le départ, la licence GPL a été rédigée dans un souci de neutralité technologique, la version 2 de la licence a mal résisté à certaines pratiques ou procédés de certains acteurs du marché.

Deux exemples ont de contournement de la licence libre ont défrayé la chronique :

- la société américaine de « câblodistribution » TIVO a utilisé des programmes à partir de briques sous licence GPL version 2 pour son magnétoscope numérique : une signature numérique empêche toute modification du code source (un contentieux très similaire oppose actuellement la communauté du libre à la société FREE pour sa FREEBOX) ;

- comme évoqué plus haut, sur le fondement de la directive communautaire du 22 mai 2001, les Etats membres de l'Union européenne ont adopté des mesures protégeant par la voie pénale, les mesures techniques de protection anti-piratage (DRM), posant la question de tels DRM conçus avec des briques de logiciel libres.

Ces évolutions législatives et certaines lacunes des premiers textes, ont provoqué le 29 juin 2007 l'adoption d'une troisième version de la GPL, seize ans après la rédaction de la version 2 de cette licence par la FREE SOFTWARE FOUNDATION. Cette nouvelle version a apporté les améliorations suivantes :

- des définitions plus approfondies des termes de la licence (sur la notion de code source, de distribution régulière ou pas...) ;
- suite au précédent TIVO, l'interdiction d'empêcher, par des dispositifs physiques ou techniques, le libre accès aux codes sources ;
- le développeur qui place son logiciel sous licence GPL v3 peut toujours déposer un brevet logiciel sur les codes sources qu'il a lui-même créé, en revanche, il s'interdit de poursuivre les utilisateurs sur le fondement des brevets logiciels ;
- la possibilité d'ajouter des restrictions supplémentaires à la licence.

Si certaines « faiblesses » de la GPL ont ainsi trouvé une réponse avec la version 3, il n'empêche pas moins que selon les cas ou le point de vue de l'utilisateur concerné, cette licence peut présenter des menaces.

Par ailleurs, la version 2 de la GPL ne peut porter que sur du logiciel. C'est pourquoi, pour la documentation accompagnant le logiciel sous GPL, elle est généralement sous licence GNU/FDL pour « Free Documentation License » (souvent abrégé en GFDL), cela n'est toutefois pas une obligation, d'autres licences peuvent être utilisées. La version 3 de la GPL permet en son article 0 de couvrir tout type d'œuvre : dans des cas très particuliers comme pour les circuits semi-conducteurs, au regard de l'article L. 622-5 du CPI qui prévoit un régime de protection très spécifique, l'application de la GPL qui repose sur le seul droit d'auteur peut sérieusement être mise en doute.

1.3.3 Menaces

Du point de vue des utilisateurs, la GPL ne présente aucune garantie de fonctionnement et, en cas de dommage, la responsabilité de l'auteur du code défectueux ne peut être recherchée (pour la version 2 : articles 11 et 12, version 3 : articles 15 et 16).

Au regard de cette absence totale de garantie, il est beaucoup plus compréhensible que des sociétés de service proposent des services de maintenance payants sur des logiciels libres.

Il est d'ailleurs possible de se demander si, en droit français, le droit de la consommation voire même le droit commun des vices cachés ne pourrait pas trouver à s'appliquer tout de même sur des logiciels libres...

La version 3 de la GPL va pourtant encore plus loin en donnant directement au juge des instructions d'interprétation de la clause d'exonération de garantie. L'article 17 prévoit que si une telle clause devait être

incompatible avec le droit local, il devrait faire application de tout mécanisme juridique qui se rapprocherait le plus d'une exonération complète de toute responsabilité civile relative aux œuvres couvertes par la licence.

Aucun cas de jurisprudence ne s'est prononcé sur la validité des clauses précitées des licences libres. La situation juridique n'est pas fixée : pour un utilisateur professionnel, le mieux est d'avoir recours à un intégrateur de la solution libre. En cas de défaillance de sa part, il sera plus aisé de mettre en cause sa responsabilité en cas d'échec du projet plutôt que de poursuivre l'un des multiples auteurs (coauteurs) du logiciel libre.

Pour le développeur qui souhaite créer sa licence libre, le recours à des briques sous licence libre est très souvent générateur de graves inquiétudes sur la protection de son-faire et de la plus-value développée à ce titre. En effet, l'effet viral de la GPL, oblige le développeur à devoir considérer d'autres voies pour préserver ses droits et les valoriser auprès de ses partenaires financiers.

1.3.4 Opportunités

La GPL est une licence jouant un formidable rôle de catalyseur de talents : le partage du code-source diffuse la connaissance et accélère le développement de fonctionnalités innovantes.

Cette dynamique de progrès implique pour les sociétés distribuant des logiciels libres de faire évoluer leur business plan en l'orientant plus directement vers les services et surtout le client.

Le fait de placer un logiciel sous licence libre implique, en effet, la renonciation à la vente d'un produit logiciel en tant que tel. Deux autres voies sont alors offertes à l'éditeur souhaitant commercialiser du logiciel libre :

- la vente de services basés sur un logiciel ;
- la vente d'une bibliothèque logicielle.

La vente de service est le premier schéma qui a été mis en œuvre par les auteurs de logiciels libres souhaitant donner une valeur économique à leur logiciel. Ils peuvent ainsi capitaliser sur leur expérience et n'ont qu'à développer leur relation client. Il faut donc prévoir un budget marketing important pour trouver la clientèle et le risque existe toujours qu'un concurrent ne reprenne l'intégralité du code source développé pour proposer ses propres services. L'existence d'une telle concurrence sauvage peut être stimulante : par exemple entre REDHAT et l'activité de MANDRAKE. Cependant, si le marché vient à être trop petit, la segmentation exagérée de ce marché sur le seul critère des services offerts peut conduire à la disparition des deux acteurs présents.

La vente d'une bibliothèque logicielle est en général commercialement viable. Toute entreprise cliente, en même temps qu'elle acquiert une suite logicielle prendra la précaution de souscrire à des services de maintenance, hotline et support. Pour certains clients, eux-mêmes éditeurs de logiciels propriétaires, le souci de ne pas voir leur logiciel « contaminé » par la licence GPL peut les conduire à demander d'acheter des licences libres garanties sans effet viral. C'est par exemple le modèle choisi par Trolltech pour Qt mais sous une autre licence « libre » moins contraignante la LGPL (voir infra). Il n'est donc pas rare de voir un même code soumis, par son auteur, à deux licences distinctes pour répondre aux besoins de ses clients.

La licence GPL est source de nombreuses opportunités tant scientifiques que commerciales. Pour pouvoir saisir au mieux ces avantages, encore faut-il mener une réflexion approfondie en amont de tout projet.

1.4 Les autres licences

Les licences libres autres que la GPL se répartissent entre celles appartenant à la famille BSD et celles que l'on peut qualifier de rares, même si elles peuvent s'affilier à l'une des deux grandes familles précitées.

1.4.1 La famille BSD

Cette famille de licences est très directement liée à de prestigieux établissements universitaires américains, tels que l'Université de CALIFORNIE, BERKELEY ou le MASSACHUSETTS INSTITUTE OF TECHNOLOGY (MIT). La grille d'analyse FFMO est également appliquée à cette famille de licences.

Forces

Le poids moral de ces établissements universitaires américains est une caution morale importante.

Pour les développeurs soucieux de distinguer un cœur de l'application logiciel (plutôt standard) donc ouvert, de développements spécifiques porteurs de valeur ajoutée, les licences BSD sont d'une grande utilité.

Le texte « officiel » de la licence BSD est très court, c'est une force, peut être autant qu'une faiblesse.

Faiblesses

Sous licence BSD, le développeur qui souhaiterait développer des adaptations ou des programmes ne doit pas tirer partie de la dénomination de l'auteur (l'Université de CALIFORNIE, BERKELEY) pour faire la publicité de leur propre produit. Les diverses contributions doivent être précisément distinguées.

Menaces

La licence type BSD est la plus ancienne, la première version remonte à 1979. Le texte de la licence peut être librement modifié par l'auteur du logiciel qui souhaite le soumettre à cette famille de licence.

Il existe donc une multitude de versions dérivées de la BSD, à quelques mots près. Cela peut être un handicap pour la compréhension de la licence applicable.

Du reste, comme pour la licence GPL, aucune garantie sur le code source ouvert n'est consentie et l'Université de CALIFORNIE n'est responsable de rien.

Opportunités

La licence BSD laisse une grande liberté dans l'utilisation du logiciel et de son code source, notamment pour les œuvres dérivées. Les travaux dérivés du code source BSD peuvent être sous licence propriétaire.

Ainsi, APPLE a basé MacOSX sur FreeBSD, MICROSOFT a repris du code TCP-IP sous licence BSD dans Windows.

Pour les entreprises ne souhaitant pas divulguer l'intégralité du code source de leur solution, la licence BSD convient idéalement pour placer, par exemple, la partie des codes la plus générique sous BSD, l'autre, sous licence propriétaire.

1.4.2 Licences spécifiques

Les licences réellement isolées sont extrêmement rares, les licences que l'on peut qualifier de « spécifiques » se rattachent, en réalité, plus ou moins à l'une des deux grandes familles de licences libres : BSD ou GPL. Régulièrement, l'OSI ou la FSF homologuent de nouvelles licences.

Les licences rares sont très nombreuses dans la famille BSD, en raison de la grande « flexibilité » autorisée par cette licence. A ce titre, il est possible de citer la licence MIT et la licence APACHE. Sous l'égide la fondation APACHE, une cinquantaine de projets emploient actuellement cette licence.

Les licences SUN (CDDL) et MOZILLA (MPL) sont également des dérivées de la licence BSD.

Dans la famille GPL, la LGPL et l'AGPL sont des alternatives soit pour infléchir les rigueurs de la GPL : c'est le cas de la LGPL, soit au contraire pour la renforcer dans son esprit *copyleft*, c'est l'AGPL.

La LGPL ou Lesser GPL

Les œuvres dérivées de la LGPL ne sont pas nécessairement soumis à cette licence. Leur auteur peut déterminer quelle licence il souhaite leur appliquer.

La possibilité de lier un programme à un autre est la force majeure de la LGPL. Cette licence est appropriée pour des bibliothèques de fonctions destinées à être appelées par différents programmes, sans trop de contraintes. Cependant, toute modification de code source dans la bibliothèque LGPL devra être également publiée sous la licence LGPL. Ce cas d'utilisation est défini par le paragraphe 5 de la licence LGPL (« *work that uses the library* »).

Un développeur peut donc lier le logiciel LGPL à un programme propriétaire qui conservera sa qualité de propriétaire, tant que le code du logiciel LGPL n'est pas modifié. Dans le cas contraire, le logiciel propriétaire est susceptible de devenir libre.

C'est pourquoi, des développeurs ont préféré diffuser leur code source sous GPL avec un avenant à la licence autorisant l'appel de fonctions par une « *special linking exception* ». Cette solution considérée comme encore plus permissive et moins contraignante a été retenue par SUN pour le JDK.

L'AGPL (AFFERO)

Le « *cloud computing* » lié aux offres commerciales SaaS (« *software as a service* ») envisage le logiciel comme un outil directement accessible au client, sans déploiement sur son système informatique.

Au sens de la GPL et même s'il y a matière à interprétation, le logiciel n'est, dans ce cas de figure, pas « distribué » ; les sources peuvent donc rester secrètes.

La société américaine AFFERO s'est émue de cette situation et a rédigé une licence éponyme : l'Affero General Public License (AGPL). Elle stipule que si le logiciel initial permet un accès par le réseau et diffuse ses sources par le réseau, alors le logiciel dérivé doit aussi distribuer ses sources.

Les différentes licences libres disponibles peuvent être classées en fonction du niveau de liberté accordé à l'utilisateur ou plutôt en fonction de l'intensité des obligations posées au regard des canons qui fondent le « mouvement libre ». La multiplication des critères en présence comme l'augmentation sensible du nombre de licences en présence rend toutefois de plus en plus difficile la classification de ces licences sur une échelle graduée. Dans le cadre, d'un projet informatique, cette question de la « gradation » des licences doit être placée au second plan : un problème beaucoup plus épineux est l'audit de compatibilité des licences sur les logiciels sélectionnés par un utilisateur sur un projet donné.

1.5 La question de la compatibilité entre différentes licences

Dans un système d'information, il faut se représenter la succession de logiciels libres employés comme une chaîne dont les maillons doivent être compatibles pour être reliés entre eux. Si différents programmes peuvent parfaitement fonctionner entre eux, tel n'est pas forcément le cas du point de vue des licences en présence : un maillon de la chaîne peut vite se briser. Sont abordées successivement les questions de compatibilité des licences GNU entre elles, puis des licences BSD entre elles, puis la confrontation de ces deux familles de licences.

1.5.1 La compatibilité entre licences GNU

La compatibilité entre les différentes versions de la GPL

Elle est prévue à l'article 9 de la version 2 et dans les mêmes termes à l'article 14 de la version 3.

Si la licence du programme utilisé stipule un numéro de version et un seul, alors seule cette version s'applique y compris pour les adaptations.

Si la licence du logiciel que l'utilisateur souhaite utiliser ne précise pas de numéro de version de GPL, alors il est possible d'utiliser n'importe quel numéro de version souhaitée.

Si la licence précise un numéro de version avec la stipulation « ou toute version supérieure », alors il est possible pour l'utilisateur de choisir la version nommée ou toute autre version postérieure.

Toutefois, la GPL version 3, article 7, en ouvrant la possibilité de modifier le texte de la licence par des restrictions supplémentaires a introduit potentiellement des problématiques de comptabilité à apprécier au cas par cas.

Même des licences supposées pouvoir être utilisées de concert ne sont pas forcément compatibles. Il en va ainsi de la licence GPL avec la licence GFDL. L'une porte généralement sur le programme informatique, la seconde sur la documentation associée. La GFDL permet d'apporter des restrictions à la modification de la documentation. Si une partie de la documentation doit être introduite dans le code source du logiciel, alors il y a incompatibilité avec la GPL qui, au moins dans sa version 2, pose le principe absolu du droit de faire évoluer les sources du logiciel. Dans ce cas, la solution est (pour l'auteur) de prévoir une double

licence sur la documentation : une licence permettant les modifications sera appliquée à la documentation, si elle doit être transposée en tout ou partie au niveau des codes sources.

La compatibilité entre licences de la famille GNU est loin d'être évidente : la solution est le plus souvent la double ou multi-licence... à condition que l'auteur l'accepte. Tel est le cas, par exemple, pour la majeure partie du code source du célèbre moteur de recherche FIREFOX, publiée à la fois sous la licence maison Mozilla Public License (MPL), la GPL et la LGPL.

1.5.2 La compatibilité entre licences de la famille BSD

Bien que généralement moins exigeantes que les licences GNU, les nombreuses versions de licence BSD sont un facteur de complexité donc d'incompatibilités.

Au sein des licences de familles BSD, seul un audit *in concreto* des licences en présence peut avoir un sens.

1.5.3 La compatibilité entre licences GNU / BSD

Quant à la compatibilité de la GPL en rapport avec les licences BSD, des tableaux de concordances figurent sur certains sites spécialisés, notamment les pages concernées sur WIKIPEDIA.

Ainsi, et sans prétendre à l'exhaustivité, les licences BSD, MIT et APACHE sont généralement considérées comme compatibles avec la GPL version 3.

Les licences non-compatibles sont SUN CDDL, ECLIPSE et MOZILLA.

En cas d'incompatibilités entre des logiciels libres, il faut se résoudre à renoncer d'utiliser un logiciel. Comme la GPL est dominante, ce sont souvent les logiciels sous d'autres licences qui doivent être écartés.

Si l'auteur du logiciel concerné l'accepte, la solution d'avoir plusieurs licences applicables à un même logiciel permet de résoudre les problèmes croissants d'incompatibilités entre licences. Même dans ce cas, l'utilisateur devra être vigilant car, sur un même logiciel, il est généralement prévu un mode distinct de rémunération de l'auteur selon la licence applicable. A défaut de solution au problème de compatibilité, l'utilisateur devra se résoudre à renoncer à l'utilisation de tel ou tel logiciel. Ces problèmes de compatibilité sont un frein réel à l'utilisation du logiciel libre, notamment pour certaines applications très orientées métier.

CHAPITRE 2

LE DROIT FRANCAIS

2.1 La paternité du logiciel libre

2.1.1 Le logiciel, une œuvre de l'esprit protégée

Le logiciel, qu'il soit « standard » ou « libre » est avant tout une œuvre de l'esprit, qui manifeste un travail intellectuel personnel de son ou ses auteurs. En tant que telle, l'œuvre bénéficie de la protection offerte par l'article L 111-1 du Code de la Propriété Intellectuelle :

« L'auteur d'une œuvre de l'esprit jouit sur cette œuvre, du seul fait de sa création, d'un droit de propriété incorporelle exclusif et opposable à tous. »

La jurisprudence a dégagé très tôt l'existence de cette protection en matière de logiciel. Ainsi, l'arrêt « PACHOT », rendu par l'Assemblée Plénière de la Cour de Cassation le 7 mars 1986, a clairement affirmé que le logiciel bénéficiait de la protection de droit commun, puisque « leur auteur avait fait preuve d'un effort personnalisé allant au-delà de la simple mise en œuvre d'une logique automatique et contraignante et que la matérialisation de cet effort résidait dans une structure individualisée ».

Dès lors qu'une œuvre manifeste cet « effort personnalisé », et qu'elle porte l'empreinte de la personnalité de son auteur, ce dernier va jouir pleinement du droit de propriété sur sa création, qui comporte deux catégories de prérogatives :

- « des attributs d'ordre patrimonial » qui, en théorie, permettent à l'auteur de vivre de sa création. L'auteur tire des bénéfices de la reproduction et de la représentation de son œuvre ;
- l'auteur jouit également d'« attributs d'ordre intellectuel et moral » au nombre desquels figurent le droit de divulgation, le droit de repentir et de retrait, le droit au respect de l'œuvre, mais également le droit à la paternité. « L'auteur jouit du droit au respect de son nom, de sa qualité et de son œuvre » (art. L 121-1 du Code de la Propriété Intellectuelle).

2.1.2 La protection de la paternité en droit commun et en matière de logiciel.

En droit commun, l'auteur peut revendiquer la paternité de son œuvre, ne pas la révéler, la retirer, ou la contester si l'œuvre a été faussement attribuée. L'auteur peut exiger la mention de son nom sur l'œuvre, exiger l'anonymat, voire utiliser un pseudonyme.

L'idée est que l'œuvre étant un prolongement de la personnalité et de l'âme de l'auteur, ce dernier doit pouvoir en faire ce qu'il entend. Cela tombe sous le sens lorsqu'on pense aux œuvres littéraires et musicales, mais paraît moins évident en matière de logiciel.

En droit du logiciel, s'agissant des logiciels « standards », le législateur a une vision différente du droit commun, qu'il a consacrée par une loi du 10 mai 1994. L'article L 121-7 du Code de la Propriété Intellectuelle, issu de cette loi, dispose que :

« Sauf stipulation contraire plus favorable à l'auteur d'un logiciel, celui-ci ne peut :

1° S'opposer à la modification du logiciel par le cessionnaire des droits mentionnés au 2° de l'article L. 122-6, lorsqu'elle n'est préjudiciable ni à son honneur ni à sa réputation ;

2° Exercer son droit de repentir ou de retrait. »

Le droit de retrait et de repentir sont déniés à l'auteur d'un logiciel, essentiellement pour des considérations pratiques, puisqu'il paraît difficile voir impossible d'effacer d'une multitude d'ordinateurs un programme que son développeur entendrait finalement conserver.

2.1.3 La paternité du logiciel libre dépend des modalités de son développement.

La philosophie du logiciel libre est la contrepartie. Cela consiste à inciter le licencié, qui bénéficie d'un accès au code-source, à communiquer les développements et à signaler les bugs afin d'en faire profiter la collectivité des internautes. L'évolution rapide du logiciel est ainsi favorisée.

Plusieurs remarques importantes sont à formuler :

- Le logiciel dit « libre » n'est pas libre de droits. Chaque contributeur peut revendiquer sa qualité d'auteur.
- Par conséquent, l'auteur a droit au respect de l'œuvre, ce qui implique l'absence d'atteinte à l'honneur ou à sa réputation.

- En outre, il faut avoir à l'esprit que les contributeurs sont multiples ; toutes les licences libres respectent le droit à la paternité !

Le respect du droit à la paternité de l'œuvre (et a fortiori du logiciel libre) impose, au préalable, d'en **identifier l'auteur**. L'élaboration d'un logiciel est toujours le fait de plusieurs personnes. Cette caractéristique est exacerbée en matière de logiciel libre, puisque des dizaines, des centaines, voire des milliers de personnes participent à son développement.

Les logiciels libres connaissent en général une approche désintéressée en matière de droit d'auteur. Le contributeur sait qu'il récoltera peu de fruits de son travail. Il s'agit souvent d'un passionné qui ne se soucie guère des conséquences financières qu'il pourrait tirer de sa participation à l'évolution de l'œuvre.

En revanche, le contributeur revendique l'« empreinte de sa personnalité » pour la partie du code-source qu'il a améliorée. La reconnaissance de sa qualité d'auteur et de l'originalité de son apport font alors ressurgir la problématique du droit à la paternité. Or, selon la nature de l'œuvre en question, le contributeur ne sera pas nécessairement qualifié d'« auteur », au sens juridique du terme. Selon que l'œuvre est collective, de collaboration ou dérivée, les titulaires du droit à la paternité diffèrent.

L'œuvre collective comprend un initiateur-directeur du projet et plusieurs contributeurs / développeurs : la propriété de l'œuvre revient par principe à cet initiateur-directeur du projet, qui détient l'ensemble des droits patrimoniaux et moraux (dont le droit à la paternité).

L'exemple-type est celui du logiciel dit « propriétaire », qui est créé à l'initiative et sous la direction d'une société ad hoc. Les droits patrimoniaux qui s'y rattachent font l'objet d'une cession plus ou moins automatique à l'employeur. Les droits moraux, en principe inaliénables, peuvent néanmoins faire l'objet d'une « cession » précaire à l'employeur (Civ 1ère, 13 février 2007). Il est donc envisageable que le véritable auteur de l'œuvre soit privé de son droit à la paternité, du moins temporairement.

L'œuvre de collaboration est celle à laquelle ont concouru plusieurs personnes physiques, qui en sont « copropriétaires », et qui peuvent chacune exploiter leur propre contribution. Dans ce cas, chaque contributeur détient un droit à la paternité sur sa part de la création. Si c'est un logiciel libre, le contributeur détient le droit à la paternité sur la partie de programme qu'il a créée.

L'œuvre composite ou œuvre dérivée est celle qui est incorporée à une œuvre préexistante. C'est surtout cette qualification que le logiciel libre est susceptible de recouvrir. Souvent, les améliorations d'un logiciel libre n'effacent pas le programme préexistant, mais s'y incorporent, pour former un nouveau programme. Le programme A va se retrouver incorporé dans un programme B, qui sera une version améliorée et portant un autre nom. Linux en est l'exemple le plus parlant.

Dans le cas d'une œuvre composite, ou dérivée, l'article L113-4 du Code de la Propriété Intellectuelle dispose que :

« L'œuvre composite est la propriété de l'auteur qui l'a réalisée, sous réserve des droits de l'auteur de l'œuvre préexistante. »

En vertu de ce texte, le droit de paternité sur le logiciel libre B ne pourra s'exercer sans l'autorisation de l'auteur du logiciel A, au titre de son droit à la paternité, et plus largement au droit au respect de l'œuvre première dont le logiciel B dépend.

C'est dans cette hypothèse que la paternité de l'œuvre revêt toute son importance : pour revendiquer la qualité d'auteur de logiciel libre, il ne faut pas être dans la position d'un contrefacteur : il faut donc détenir une licence pour faire évoluer ce logiciel libre.

L'intérêt des licences de logiciel libre est donc de définir les règles de contribution et de dévolution des droits patrimoniaux et moraux, souvent dérogoratoires au droit commun. Les licences les plus courantes sont :

- les licences « copyleft », où l'auteur de l'adaptation s'engage à ce que son adaptation soit elle-même libre de droits ;
- la licence GPL où les adaptations du logiciel n'appartiennent pas au développeur, qui ne peut donc pas transmettre les droits sur l'adaptation à son client.

2.2 L'œuvre dérivée et le logiciel libre.

La notion d'œuvre dérivée, évoquée précédemment, peut poser problème quand on aborde les licences libres, et notamment la licence GPL que nous étudierons en détail dans le chapitre suivant.

En effet, l'aspect viral de la licence GPL implique qu'une œuvre dérivée d'un logiciel soit redistribuée sous la GPL. Or cette notion est difficile à appréhender en matière de programmes informatiques.

Pour déterminer les œuvres couvertes par la licence GPL, le texte de la licence GPL fait référence au "*derivative or collective work based on the program*" et au "*work derived from the program*" (cette dernière expression n'est pas reprise dans la v3) ... en les opposant aux "*independent and separate works in themselves*" censés ne pas être contaminés par la licence.

Dans la traduction non officielle de la licence, les rédacteurs ont utilisé la notion de "*travail dérivé ou collectif basé sur le Programme*".

Mais il est nécessaire de déterminer ce qu'est une œuvre dérivée en matière de logiciel. La communauté du logiciel libre n'est pas unanime sur la question et de nombreux débats existent à ce sujet. Globalement, il est fait une distinction entre :

- le "*static linking*" : le code de la librairie issu du logiciel initial est repris dans la version exécutable finale du nouveau logiciel.
- le "*dynamic linking*" : le code de la librairie issu du logiciel initial n'est pas incorporé dans la version exécutable finale du nouveau logiciel.

L'interprétation extensive de la licence GPL, qui est celle de ses auteurs, prévoit qu'une œuvre est dérivée en cas de "static linking" ou de "dynamic linking" tandis que certains spécialistes estiment qu'en cas de "dynamic linking", l'œuvre n'est pas dérivée car elle ne doit pas être modifiée pour que le "*linking*" s'effectue.

Au regard des enjeux économiques et de la difficulté de compréhension du débat pour les non-spécialistes, il apparaît normal que le champ d'application des licences libres ne soit pas encore tout à fait déterminé. Comme nous le verrons après notre étude de la licence GPL, ceci a posé récemment quelques problèmes aux tribunaux français.

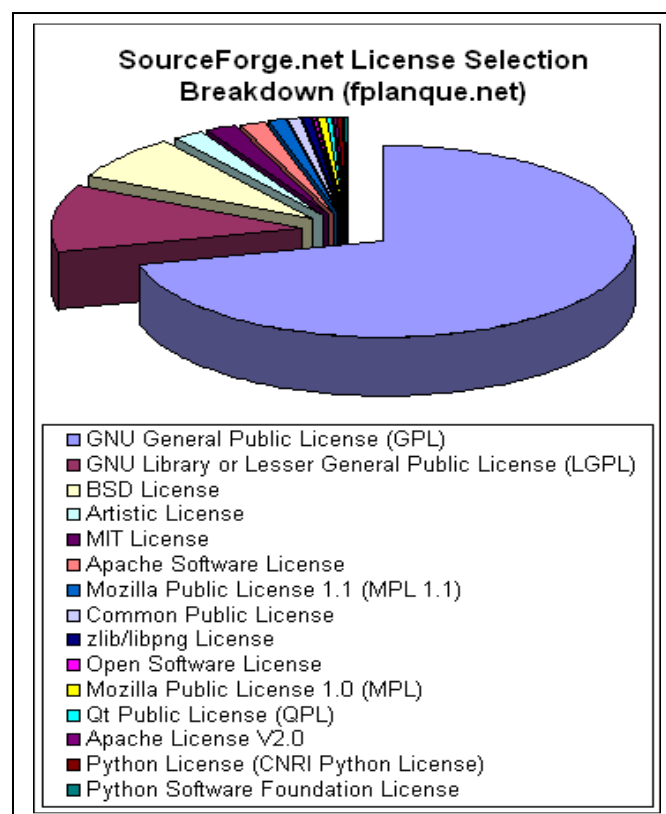
CHAPITRE 3

LICENCE GPL

Afin de permettre la distribution de programmes exempts de droits, la fondation FSF (*Free Software Foundation*, traduisez *Fondation pour les logiciels libres*), www.fsf.org fondée en 1980 par Richard Stallman, a développé un projet nommé GNU («GNU is Not Unix»).

Les utilitaires GNU sont soumis aux termes de la licence d'utilisation **GPL** (*General Public License*) décrivant les conditions légales de l'utilisation, de la distribution ou la modification du code source.

La licence GPL est associée à près de 70% des logiciels libres du marché. La version la plus récente est actuellement la V3 qui a été promue par la FSF le 29 juin 2007.



3.1 Les principaux aspects de cette licence sont que :

- la licence GPL permet la modification du programme original, et sa diffusion (sous licence GPL) ;
- la licence GPL autorise la vente du logiciel libre sous sa forme originale ou modifiée, à condition que le vendeur autorise la diffusion (même gratuite) du logiciel ainsi modifié ;
- la licence GPL autorise l'utilisation du logiciel à des fins lucratives ;
- les logiciels sous licence GPL restent la propriété de leurs auteurs, personne ne peut donc s'approprier tout ou partie des droits d'auteur ;
- la licence n'implique aucune forme de rémunération des auteurs.

Pour résumer, les quatre conditions exigées par la FSP sont l'accès au code source du programme, la possibilité de modification et d'adaptation du programme, la possibilité de redistribution des copies du programme et l'obligation de publication du code source modifié.

Pour définir des conditions d'utilisation supplémentaires, la GPL place le programme qu'elle protège sous "copyleft". Cette licence est parfois appelée copyleft, par analogie avec copyright, "droits d'auteur" en français (*right*=droit et *left*=laissé, libre). Ce principe oblige que le code source modifié doit être à son tour placé sous GPL pour éviter qu'un programme sous GPL ne devienne un jour un programme propriétaire.

Certains éditeurs informatiques utilisent un dérivé de la GPL pour la distribution de leurs produits, la licence Open Source.

Ces logiciels sont accompagnés de leur code source, mais il n'y a pas d'autorisation à modifier le code source ni à redistribuer le programme sans l'accord de l'éditeur. Cette licence autorise les modifications et les travaux dérivés, et leur distribution sous les mêmes conditions que celles autorisées par la licence du programme original.

La GPL marque donc l'empreinte des logiciels libres par ses spécificités et par son mode de distribution notamment.

3.2 Distribution et GPL

3.2.1 Distribution et codes sources

La GNU General Public License pose des exigences en ce qui concerne la distribution des modifications apportées à un logiciel publié sous cette licence.

En effet, la licence GNU impose aux développeurs de rendre public le code source des modifications du fait de leur distribution.

Une erreur souvent faite est celle liée à l'interprétation des obligations qu'impose la GPL à un développeur apportant des modifications à un logiciel libre, la plus commune étant de clamer qu'il est dans l'obligation de publier ses modifications.

Cette licence autorise les utilisateurs à obtenir et étudier le code source d'un logiciel libre. Ils peuvent également le modifier pour l'adapter à leurs besoins comme l'indique le

second article du paragraphe "TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION" de la licence : art 2. « *You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program* ».

Mais rien n'oblige le développeur, premièrement, à distribuer ses modifications et, deuxièmement, à les placer sous licence GPL tant qu'il ne les distribue pas. Cette question fait l'objet d'ailleurs l'objet d'une réponse dans la FAQ officielle de la GPL :

« Does the GPL require that source code of modified versions be posted to the public?

The GPL does not require you to release your modified version. You are free to make modifications and use them privately, without ever releasing them. This applies to organizations (including companies), too; an organization can make a modified version and use it internally without ever releasing it outside the organization. But if you release the modified version to the public in some way, the GPL requires you to make the modified source code available to the program's users, under the GPL.

Thus, the GPL gives permission to release the modified program in certain ways, and not in other ways; but the decision of whether to release it is up to you ».

Pour traduire simplement, la réponse apportée énonce que tant qu'un développeur ne distribue pas, il peut modifier et utiliser le résultat à sa convenance et dans les conditions qu'il souhaite, qu'il soit une personne physique ou une personne morale (une organisation, une société...).

Ainsi, l'utilisation et le déploiement d'un programme au sein d'une même entreprise ne constitue pas a priori une

distribution. Cela engendre le fait qu'une personne morale peut créer une œuvre dérivée en l'utilisant en interne, sans limitation du nombre d'installations (postes ou serveurs), sans être tenue de diffuser les codes sources de l'œuvre.

Maintenant, si un développeur a décidé de distribuer le résultat de son travail, le même article en précise les conditions : art 2. « *You may [...] copy and distribute such modifications or work under the terms of Section 1 above* ».

Ainsi, la distribution des modifications doit se faire sous licence GPL. En revanche, il n'y a pas d'obligation de publication, dans le sens de mise à disposition, des modifications distribuées. S'il doit fournir ses travaux dérivés sous GPL au public à qui il les distribue, le développeur n'a en revanche aucune obligation envers le public non visé. La FAQ précédemment mentionnée répond elle aussi à cette question :

« *But if you release the modified version to the public in some way, the GPL requires you to make the modified source code available to the program's users, under the GPL* ».

Le développeur distribue son programme qui, même s'il est le résultat de modification d'un autre programme, n'en reste pas moins un programme différent dont les modifications lui appartiennent.

Mais la distribution d'un logiciel sous licence GPL ne permet pas de maîtriser sa diffusion ultérieure puisque la distribution entraîne le droit de redistribuer librement le programme. La distribution, même simple, met donc potentiellement le logiciel modifié à disposition de tout le monde. Ainsi, le développeur devra considérer que rien

n'empêchera un de ses destinataires de redistribuer lui-même le logiciel, voire de le publier purement et simplement.

En résumé, une personne ne peut obliger le développeur à publier le code source d'une modification s'il ne lui distribue pas la modification en question, et a fortiori s'il ne la distribue pas du tout. Mais l'absence de maîtrise dans la distribution rend le problème plus complexe et montre la limite de la liberté laissée aux développeurs.

3.2.2 Distribution et transmission de licence

En accord avec les principes du développement coopératif, les développeurs peuvent rediffuser le logiciel mais pour être sûrs de continuer à protéger le logiciel par la GPL, il leur est interdit de le faire par le biais d'une sous-licence.

L'article 2 de la GPL prévoit ainsi que le développeur doit adresser à tout destinataire du logiciel un exemplaire de la licence GPL ou lui indiquer comment y accéder.

Ainsi, chaque fois que le développeur distribue le logiciel, modifié ou non, le destinataire reçoit directement du donneur de licence les droits de dupliquer, distribuer et modifier le logiciel selon les termes de la licence.

C'est toujours le donneur de licence initial qui continuera à occuper la place du donneur de licence au fur et à mesure que de nouveaux partenaires s'impliqueront dans le développement coopératif du logiciel.

Cette solution très originale permet de garantir la stabilité de l'édifice GNU autour du droit de l'auteur original du logiciel.

3.3 Sur l'aspect viral de la GPL

Selon l'article 2 de la GPL, « *You must cause any work that you **distribute** or **publish**, that in whole or in part contains or **is derived** from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this **License*** ». Ainsi, si un développeur crée un logiciel dérivé du logiciel original et dépend de celui-ci, alors il sera dans l'obligation de distribuer son travail sous GPL.

De plus, selon l'article 3 de la GPL, le développeur qui souhaite distribuer à son tour le logiciel doit d'abord indiquer les éventuelles modifications qu'il y a apportées pour que les autres licenciés de la communauté puissent en bénéficier et que les améliorations apportées puissent être intégrées à la version suivante du logiciel.

De cette façon, les logiciels dérivés d'un logiciel GPL sont eux-mêmes des logiciels GPL.

C'est Steve Ballmer, PDG de Microsoft, qui a qualifié cette particularité d'aspect « viral » de la GPL. On parle également de réciprocité, ou de donnant-donnant. En effet, le développeur peut devenir donneur de licence et la communauté des utilisateurs du logiciel libre, par conséquent s'agrandit, objectif à atteindre pour tout système de création coopératif.

Cette extension de la GPL a ses limites indiquées précédemment aux articles 3-2 et 3-3 : le développeur a le droit d'exploiter séparément ses modifications (voir paragraphe « distribution et codes sources ») et d'utiliser du code GPL dans une œuvre sous licence différente à

condition qu'elle soit raisonnablement indépendante et distincte.

Ainsi, l'aspect viral de la GPL peut être contourné en s'abstenant de distribuer le logiciel ou bien de créer un logiciel indépendant qui ne caractérise pas une œuvre dérivée.

Il convient de préciser ce dernier point.

3.4 GPL et œuvre dérivée

La licence GNU GPL, « GNU General Public Licence », se caractérise donc principalement par le droit de modifier le programme et de redistribuer ces modifications, qui constituent des œuvres dérivées, à la condition que ce soit sous la même licence GPL.

C'est, comme nous l'avons vu, le caractère viral de la licence qui se communique aux travaux dérivés. Il convient donc de déterminer la nature et le régime d'une œuvre dérivée.

Une œuvre dérivée peut s'analyser par la modification d'un morceau du code source d'un programme A pour obtenir un programme B. De même, si la modification tient dans l'appel des fonctions du programme A depuis un programme B, en liant les deux programmes (« link »), c'est également une œuvre dérivée. Cette liaison entre les programmes peut être statique ou bien dynamique (exécution seule ou non). Comme nous l'avons vu dans le chapitre 2 de ce livre blanc, il existe encore aujourd'hui un débat sur la question de savoir si une liaison dynamique permet la création d'une œuvre dérivée.

De manière plus large, il est admis qu'un logiciel B est considéré comme une œuvre dérivée du logiciel A, si B ne peut pas fonctionner de manière utile sans A, ceci indépendamment des modalités techniques de la liaison. En effet, il est certain que si un logiciel est issu d'un logiciel précédent, il a juridiquement la qualité d'œuvre dérivée.

Ainsi, en théorie, puisqu'il est dépendant du programme original, protégé par le droit d'auteur, le programme dérivé ne pourrait pas être exploité sans l'accord du titulaire des droits du programme original.

Mais, la GPL engendre des œuvres complexes par l'apport d'une multitude de contributeurs pouvant être qualifiées d'œuvre de collaboration selon une partie de la doctrine, et notamment Mme Clément-Fontaine qui affirme qu'« il y a collaboration chaque fois que les participants à une œuvre suivent une orientation donnée (une inspiration commune) qui définit le projet auquel ils participent » (Clément-Fontaine M., *les œuvres libres*, Thèse Montpellier, 2006).

3.5 Sur la validité française de la GPL

La GPL en France est rédigée en anglais, et la traduction française n'est fournie qu'à titre indicatif et non-officiel.

Selon une partie de la doctrine, cette absence la disqualifierait d'emblée en tant que contrat passé entre l'utilisateur et le fournisseur du logiciel. Une autre partie estime que le cœur de cette licence ne répond pas sur certains points du code de la propriété intellectuelle mais utilise en revanche des mécanismes classiques de droit international privé.

Ainsi, de nombreux problèmes existent quant à la détermination de la valeur juridique de la GPL sur le territoire français d'une part, et de la loi applicable d'autre part.

Il est enfin à noter, à titre subsidiaire, qu'une licence française existe pour les logiciels libres appelée CeCILL spécialement rédigée en français pour le droit français et compatible avec la GPL.

CHAPITRE 4

LA JURISPRUDENCE

La licence GPL, de par sa rédaction et ses spécificités, laisse ouvertes de nombreuses questions d'ordre juridique. Ces questions ont été posées aux juges nationaux de différents Etats à l'occasion de litiges portant sur l'application des dispositions de la licence GPL.

Ce sont les juridictions américaines (4.1) qui ont, les premières, consacré la validité de la licence GPL avant d'être imitées par leurs homologues allemandes (4.2) et, très récemment, par leurs homologues françaises (4.3).

4.1. Jurisprudence américaine

De nombreux litiges étaient survenus aux Etats-Unis autour de la valeur juridique du logiciel libre mais les parties préféraient toujours régler leurs litiges hors des tribunaux. Néanmoins, quelques décisions sont venues consacrer explicitement la validité des licences de logiciel libre, en l'occurrence celle de la licence GPL.

4.1.1 District Court of Massachusetts, Progress Software Corp. v My SQL AB, 28 février 2002

Le litige entre *My SQL AB* et *Progress Software Court* a été réglé par les parties de manière amiable. Précédemment, un tribunal américain (District Court of Massachusetts, Progress Software Court v My SQL AB) avait été saisi en référé par My SQL, développeur d'un [système de gestion de base de données](#) sous licence libre dont le logiciel et la marque avaient été repris dans un logiciel propriétaire (Gemini) de la Progress Software Corp. Le tribunal a eu

l'opportunité de reconnaître la validité de la licence GPL à cette occasion. La reconnaissance n'est pas claire mais elle semble être affirmée de manière implicite dans la phrase "*with respect to the general public license*". Les juges ont cependant refusé de statuer en faveur de MySQL, aux motifs que la société n'avait pas démontré l'existence d'un dommage irréparable et qu'au moment du prononcé de la décision, la Progress Software Corp. avait mis le code source de son logiciel Gemini à disposition des utilisateurs, se mettant ainsi en conformité avec la licence GPL.

4.1.2 Wallace v. International Business Machines Corp. et al., 20 mars 2006 et 16 mai 2006

Deux affaires vont consacrer la compatibilité des logiciels libres avec le droit de la concurrence américain. Un individu, dénommé M.Wallace, avait développé son propre système d'exploitation en logiciel propriétaire. Il considérait que la gratuité du système d'exploitation sous licence GPL dénommé Linux lui empêchait de vendre son propre logiciel propriétaire créant ainsi une pratique anti-concurrentielle. Il s'est retourné d'abord contre la Free Software Foundation puis contre des sociétés incorporant le logiciel Linux à leurs produits :

- le 20 mars 2006, un tribunal de l'Indiana débouta M.Wallace de sa demande au motif qu'il n'a pas prouvé que la licence GPL causé un préjudice au marché des systèmes d'exploitation et a affirmé que "*le GPL encourage la libre concurrence et la distribution des systèmes d'exploitation, et les bénéfices se répercutent sur les consommateurs*". La licence GPL est compatible avec les visées du Sherman Act, acte fondateur du droit de la concurrence américain.

- le 16 mai 2006, un tribunal de l'Indiana différemment composé est arrivé à la même conclusion. Dans l'affaire Wallace v IBM Corp. Red Hat Inc. & Novell Inc et le 9 novembre 2006, une cour d'appel a confirmé la décision de première instance.

4.1.3 Jacobsen v. Katzer, 13 août 2008

La validité de la licence GPL a été clairement reconnue par une décision de la Cour d'appel du tribunal fédéral américain dans l'affaire Jacobsen v Katzer du 13 août 2008. Jacobsen reprochait à Katzer d'utiliser le logiciel libre qu'il avait développé dans une application propriétaire et de l'avoir breveté.

La décision a permis de clarifier la situation du logiciel libre aux Etats-Unis car les juges affirment que la licence GPL est valide et que ses dispositions sont applicables ("*enforceable copyright conditions*"). Le respect des termes de la licence est présenté comme une condition obligatoire de la concession de licence du logiciel ("*the rights to copy, modify, and distribute are granted provided that the conditions are met*").

4.2. Jurisprudence allemande

4.2.1 District Court of Munich I, 19 nov 2004

En Allemagne, la validité de la licence GPL avait été reconnue par une décision de justice de 2004. Le plaignant était Harald Welte, créateur du logiciel NETFILTER / IPTABLES et fondateur du projet gpl-violations.org, qui avait assigné la société SITECOM devant la District Court of Munich au motif qu'elle distribuait le logiciel sans les codes sources de celui-ci.

Le tribunal allemand a exigé, dans une décision du 19 mai 2004, la suspension de la diffusion des produits SITECOM intégrant le logiciel jusqu'à ce que la société se mette en conformité avec les dispositions de la licence GPL (ce qu'elle fera la semaine suivante).

La validité de la licence GPL est explicitement affirmée par le tribunal.

4.2.2 District Court of Munich, Welte vs. Skype Technologies SA, juin 2007

Une affaire similaire a eu lieu 3 ans plus tard. Harald Welte avait poursuivi cette fois-ci Skype, qui vendait un téléphone fonctionnant sous Linux, sans diffuser le code source. Harald Welte affirme sur son site internet que le tribunal de première instance allemand a statué en faveur de gpl-violations.org par un jugement de juin 2007 et que Skype a introduit un flyer dans la boîte du téléphone contenant les instructions pour se procurer le code source sur Internet. Skype a interjeté appel devant la Cour d'Appel de Munich, arguant du fait que la licence GPL serait contraire au droit de la concurrence. Après une audience devant le Oberlandesgericht Muenchen le 8 mai 2008, Skype a décidé de se plier à la première décision.

4.3. Jurisprudence française

4.3.1 TGI Paris, 28 mars 2007, Educaffix c/ CNRS, Université Joseph Fournier et autres

La société Educaffix a conclu, avec le CRNS et des établissements d'enseignement supérieur (dénommés l'Organisme dans l'arrêt), un contrat de cession de droits

patrimoniaux sur un logiciel dénommé Baghera. Educaffix souhaitait développer son propre logiciel à partir de Baghera.

Or, Baghera ne pouvait fonctionner sans JAT Lite, un logiciel créé par l'Université américaine de Stanford, distribué sous licence GPL, que l'Organisme avait légèrement adapté. L'Organisme a donc fourni, sur le Cdrom contenant Baghera, la brique de logiciel libre JAT Lite nécessaire pour le fonctionnement de Baghera mais n'a pas transmis une copie de la licence GPL.

La société Educaffix a assigné l'Organisme devant le Tribunal de Grande Instance de Paris pour demander la nullité du contrat de cession pour dol car l'Organisme lui a volontairement caché l'existence de JAT Lite qui nécessiterait l'autorisation de l'Université de Stanford.

L'argument du dol est écarté par les juges car le logiciel libre JAT Lite n'était pas inclus dans le contrat de cession. Ainsi, Educaffix ne peut pas reprocher à l'Organisme de lui avoir caché l'existence de ce logiciel.

Subsidiairement, Educaffix demande la résolution du contrat aux torts de l'Organisme parce que l'exploitation de Baghera implique nécessairement la commission d'acte de contrefaçon de JAT Lite (utilisation sans l'autorisation de l'Université de Stanford).

Les juges essaient d'appréhender la notion de logiciel libre. La licence GPL possède un caractère "contaminant" (art 2 de la GPL) qui connaît une exception quand les éléments dérivés du programme peuvent être considérés comme raisonnablement indépendants. En l'espèce, les juges estiment que Baghera a besoin de JAT Lite pour

fonctionner, et qu'il est impossible de remplacer JAT Lite efficacement.

Par conséquent, Educaffix et l'Organisme auraient dû solliciter une licence spéciale auprès de l'Université de Stanford ou remplacer JAT Lite.

Au regard des éléments du dossier, le tribunal décide de prononcer la résolution du contrat aux torts partagés des deux parties.

Cette décision a été présentée comme la première jurisprudence française sur l'utilisation d'un logiciel libre sous licence GNU GPL dans un logiciel propriétaire et sur les risques juridiques qui pèsent sur une future cession.

La reconnaissance de la validité de la licence GPL et son application à la lettre sont les deux principaux enseignements à tirer de ce jugement.

Cette décision reconnaît la validité et l'opposabilité de la licence GPL en droit français. Un développeur souhaitant diffuser son programme sous licence GPL sait que les tribunaux pourront faire valoir son droit au respect de licence et garantir la diffusion sous la même licence des œuvres dérivées du logiciel.

Le second constat est que les juges ont appliqué à la lettre le texte de la licence en reprenant à leur compte la notion de logiciel dérivé ou indépendant du logiciel libre (art 2).

Néanmoins, le tribunal n'est pas clair dans son interprétation de la licence. Cela transparait notamment quand ils considèrent qu'il aurait fallu "*solliciter une licence spéciale auprès de l'université de Stanford*" car Baghera n'était pas indépendant de JAT Lite.

Educaffix avait sollicité auprès du tribunal que celui-ci désigne un expert afin de déterminer si Baghera était indépendant ou dérivé de JAT Lite. Le tribunal a repris à son compte le rapport de l'expert qui reste flou dans son appréciation du caractère dérivé. L'expert considère que "*Baghera en tant que tel ne dérive pas techniquement de JAT Lite*" mais énonce que "*la technologie Multi-agents agents[...] ne peut être considérée comme indépendante puisqu'elle est bâtie en utilisant les objets constitutifs de la bibliothèque JAT Lite*". Le tribunal estime que Baghera dérive de JAT Lite alors qu'il est indépendant de celui-ci selon l'expert.

De plus, en estimant qu'il fallait une licence spéciale de l'Université de Stanford pour concéder des droits sur JAT Lite, les juges raisonnent comme si JAT Lite était un logiciel sous licence propriétaire. Il n'est fait aucune référence à une "*licence spéciale*" dans les termes de la licence GPL. Baghera, qui dérive de JAT Lite, doit être distribué sous licence GPL.

4.3.2 TGI Chambéry, Société ERN c/ Conseil Général de Savoie et Université de Savoie, 15 nov. 2007

Les faits de cet arrêt sont très complexes et il convient de les synthétiser en ne conservant que la partie du litige concernant les logiciels libres.

L'Université de Savoie et le Département de Savoie ont, dans le cadre d'un contrat de partenariat signé le 27 février 2004, concédé à la société ERN des droits patrimoniaux sur un portail pédagogique dénommé Cartable Electronique. La société a, par la suite, reproché aux concédants de ne pas avoir livré les codes sources nécessaire au développement technique du produit (ils ne les ont livré que 6 mois après)

et de ne pas avoir garanti l'exclusivité contractuellement promise dans la mesure où le portail était principalement développé à partir de logiciels libres.

Dans leur décision, les juges ont choisi de prononcer la résolution du protocole d'accord transactionnel (qui avait été signé par les parties et devait prévoir la signature d'un nouveau contrat) et la résiliation du contrat de partenariat seront prononcés aux torts partagés du demandeur et des défendeurs et de débouter les deux parties de leurs demandes financières.

Le tribunal va constater que les codes source du logiciel, bien que non fournis, avaient été déposés à l'APP en janvier 2003 et que, par conséquent, la société ERN pouvait librement y accéder.

La question nous intéressant plus particulièrement ici est celle de la présence de logiciels libres et des conséquences de cette présence sur l'obligation d'exclusivité.

En effet, parmi les composants du Cartable Electronique figuraient des logiciels dits libres. Le Cartable Electronique était composé de deux répertoires, dont un était constitué de logiciels libres sous licences GPL et l'autre de logiciels propriétaires. Il a été établi que la plate-forme Cartable Electronique ne peut fonctionner sans le répertoire contenant les logiciels libres.

Les juges, dans leur raisonnement, vont alors adopter la définition du logiciel libre énoncée par la Free Software Foundation. Ils vont également constater l'aspect "contaminant" de la licence GPL : les œuvres dérivées d'un logiciel sous licence GPL doivent être diffusés sous cette même licence, sous peine pour le contrevenant de perdre

l'intégralité des droits qu'il dispose sur le logiciel. En concédant un produit composé de logiciels libres sous licence GPL, les défendeurs n'auraient pas la faculté juridique de concéder des droits qu'ils n'avaient pas et, a fortiori, concéder une exclusivité sur le cartable électronique.

Les juges admettent donc explicitement que "*le principe de la licence libre ne peut être compatible avec la notion d'exclusivité puisque ce type de licences repose sur le partage de connaissances*".

Cependant, le tribunal affirme qu'il y a manquement des concédants à leur obligation d'exclusivité dès lors qu'ils concèdent une licence exclusive d'exploitation sur un programme composé en partie de modules exploités sous forme de licence libre.

Cet arrêt est plus clair que la décision Educaffix rendue quelques mois plus tôt. Les juges se sont encore appuyés sur le rapport d'un expert mais ils ont choisi d'insérer une définition du logiciel libre (en l'occurrence celle de la Free Software Foundation), de distinguer les licences *copyleft* et les licences *non copyleft*, et de mentionner l'aspect contaminant de certaines licences libres.

D'un point de vue juridique, l'apport principal de cette décision est l'affirmation de l'incompatibilité entre une exploitation exclusive et un logiciel libre. Pour le TGI, "*le principe de la licence libre fondé sur un partage de connaissances s'avère incompatible avec la notion d'exclusivité*". Les entreprises doivent faire attention lorsqu'elles concèdent une exclusivité sur un logiciel propriétaire à ce que celui-ci ne soit pas composé en partie de logiciels libres, sinon l'engagement d'exclusivité "*perd*

toute portée" (les juges considèrent que la présence d'une licence GPL modifie l'économie du contrat).

Néanmoins, le TGI entretient une incertitude quant à la distinction entre logiciel dérivé et logiciel indépendant. Il n'y fait aucune allusion alors qu'il aurait été intéressant de définir le logiciel indépendant, ce qui aurait permis de déterminer à quel moment la licence GPL cesse de s'appliquer.

On peut en déduire deux hypothèses :

- soit les juges abandonnent une telle distinction et considèrent que dès qu'un logiciel sous licence GPL a servi au développement d'un logiciel propriétaire, il n'y a pas de possibilité de concéder une licence exclusive,
- soit les juges estiment que le fait qu'un logiciel sous licence GPL soit nécessaire pour faire marcher le logiciel propriétaire est suffisant pour considérer que la licence GPL a "contaminé" ce dernier qui en est une œuvre dérivée.

4.3.3 Assignation de la S.A.S. Free devant le Tribunal de Grande Instance de Paris, novembre 2008

Free a intégré dans sa Freebox des logiciels libres, tels que Netfliters, Iptables et Busybox, qui sont distribués sous licence GPL. Or, selon les plaignants, Free ne respecte pas les termes de la licence puisqu'il ne distribue pas de copie de celle-ci et qu'il ne permet pas aux utilisateurs d'accéder au code source des logiciels libres.

Les plaignants sont au nombre de trois et sont connus pour leurs actions en justice pour la défense des logiciels libres.

On retrouve Harald Welte, ce développeur allemand à l'origine du projet NETFILTER / IPTABLES, qui avait déjà poursuivi de grosses sociétés abusant du logiciel libre sans respecter les termes de la licence (Skype etc....). Harald Welte est le fondateur du projet gpl-violations.org et a un blog où l'on peut suivre les évolutions de ses actions en justice.

Les deux autres plaignants sont deux américains, MM. Andersen et Landley, développeurs de Busybox, et ont déjà agi devant les tribunaux américains pour l'utilisation de leur programme par des sociétés qui se refusaient à publier le code source (il n'y a pas de décisions concernant ces affaires puisque elles ont toutes fait l'objet d'une transaction entre les parties).

Free joue, auprès du grand public français, de son image de promoteur du logiciel libre. La société a bâti son succès par des tarifs attractifs dus à une grande utilisation de l'open source leur permettant de limiter leurs coûts. Free dispose d'une équipe de programmeurs chargés de développer les logiciels libres en interne et de les adapter aux exigences de leur secteur d'activité. Free n'a pas à investir dans des logiciels propriétaires, ni à créer lui-même un logiciel libre : ils bénéficient amplement du travail de développeurs indépendants tel que celui effectué par les plaignants.

Les plaignants demandent que leurs droits patrimoniaux et leur droit moral soient respectés par la société Free. Ainsi, ils souhaitent que leur nom soit inscrit sur les logiciels ou, à tout le moins, sur les documents les accompagnant. Ils demandent également que Free se mette en conformité avec la licence GPL en mettant le code source et une copie de la licence à la disposition des Freenautes.

De plus, ils espèrent la condamnation de Free à leur verser des dommages et intérêts à hauteur de 1€ par Freebox mise à disposition des utilisateurs et une publication de la décision.

La décision qui sera rendue dans cette affaire, si les parties ne décident pas de transiger, sera sûrement très intéressante et donnera le cap à suivre pour les tribunaux en matière de respect de la licence GPL. La notoriété de Free et leur attitude hostile à cette assignation ont donné à cette affaire une médiatisation importante et la position du TGI de Paris sera abondamment commentée.

Free estime qu'elle ne vend pas ses Freebox à ses clients et qu'elle les met simplement à leur disposition : les Freebox restent donc "*la propriété de Free*". La licence GPL prévoit, dans sa section 0, qu'elle ne s'applique pas pour les activités autres que "*copying, distribution and modification*". La traduction de la notion de "*distribution*" pose des problèmes : les plaignants estiment que cela équivaut à une "*mise sur le marché*" en droit français et Free considère que cela équivaut à une "*distribution*". Free mettant à disposition ses Freebox qui constituent une partie de son réseau, elle ne les distribue pas et n'a donc pas à respecter les termes de la licence GPL.

Que pouvons-nous attendre de la décision des juges du fond ?

Ils vont probablement s'inscrire dans la logique des tribunaux précédents et reconnaître la validité de la licence GPL. La question la plus délicate sera celle de son interprétation, où nous avons constaté que les tribunaux français ont été maladroits pour le moment. Les juges devront déterminer si le terme anglais de distribution

correspond également à une distribution en France, comme le soutient Free, ou à une mise sur le marché, comme le soutiennent les plaignants. Une décision en faveur de l'argumentation de Free ouvrirait une faille permettant de contourner les termes de la licence en cas de mise à disposition mais enfermerait la société dans son schéma actuel de distribution des Freebox et toute modification de celui-ci (vente, location etc.) l'obligerait à respecter les termes de la licence GPL.

Il est à noter qu'il est probable que les juges appliquent pour la première fois à des créateurs de logiciels libres la notion de droit moral de l'art L.121-1 du CPI et plus particulièrement le droit au respect de la paternité de l'œuvre. Cette notion a déjà été de nombreuses fois appliquée à une œuvre logicielle mais c'était dans le cadre d'un logiciel propriétaire.

D'un point de vue économique, l'enjeu est différent. Si Free ne souhaite pas mettre à disposition le code source des logiciels sous licence GPL, c'est qu'ils ont été en partie modifiés par les programmeurs Free et que la société ne souhaite pas transmettre les évolutions apportées à ces logiciels au grand public. Une décision en faveur des plaignants rabaissera Free au rang d'un développeur Open Source lambda travaillant pour la communauté dans son ensemble et non pour les seuls intérêts de ses clients.

Le TGI de Paris a donc la possibilité de rendre une décision historique, en consacrant de nouveau la validité de la licence GPL et en permettant une application extensive de celle-ci. Si certains points resteront toujours de la licence resteront toujours à éclaircir (notion d'œuvre dérivée / œuvre indépendante), cette décision pourrait donner une

impulsion en faveur de décisions respectueuses des licences libres.

ANNEXES

- 1. Jugement TGI PARIS – 28 mars 2007 –
EDUCAFFIX**
- 2. Jugement TGI CHAMBERY – 15 novembre 2007 –
Sté ERN**
- 3. Assignation de SAS FREE devant le TGI PARIS –
novembre 2008**