



Université Toulouse 1  
Mémoire de Master 2 Propriété Intellectuelle  
Parcours techniques contractuelles

**« La protection du logiciel :  
logiciels libres vs logiciels open source ? »**

Réalisé par Sophie Virard

Sous la direction de  
Madame Mendoza Caminade Alexandra  
Maître de conférences

Année universitaire 2009-2010



## Sommaire

<b><i>Introduction</i></b>	<b>5</b>
<b><i>Partie 1 – Logiciel libre et open source : une analogie marquée</i></b>	<b>14</b>
<b>Chapitre 1 – Un concept commun</b>	<b>14</b>
Section I – Une alternative au logiciel propriétaire	15
Section II – Une possibilité réalisée par le mécanisme contractuel	26
<b>Chapitre 2 – Des outils juridiques similaires</b>	<b>37</b>
Section I – La diversité des licences	37
Section II – Une diversité partagée par les deux mouvements	47
<b><i>Partie 2 – Logiciel libre et open source : une analogie imparfaite</i></b>	<b>57</b>
<b>Chapitre 1 – L’opposition terminologique</b>	<b>57</b>
Section I – Le problème du terme <i>free</i>	58
Section II – Le problème du terme <i>open source</i>	69
<b>Chapitre 2 – L’opposition idéologique</b>	<b>78</b>
Section I – Un désaccord stratégique	78
Section II – Un désaccord pratique	90
<b><i>Conclusion</i></b>	<b>104</b>
<b><i>Annexes :</i></b>	<b>108</b>
<b><i>Bibliographie :</i></b>	<b>115</b>
<b><i>Table des matières :</i></b>	<b>120</b>

*Tant aujourd'hui la liberté sommeille  
Qu'elle ne trouve parfois de sursaut  
Que dans un cœur indigné de trop  
Alors on sait qu'elle se réveille.*

Thomas MOORE

# Introduction

1 - C'est du fait de l'indignation d'un homme, Richard Matthew Stallman, que va naître le concept de logiciel libre.

Auparavant, la plupart des logiciels étaient libres, c'est-à-dire qu'il était possible de lire le code source, de procéder à des améliorations et de diffuser des copies<sup>1</sup>. Le *hardware*, l'ordinateur en lui-même qui à l'époque était une grosse machine coûteuse, constituait le cœur de l'économie informatique. Quant aux logiciels, on se souciait peu de leur protection dans la mesure où ils n'avaient pas une grande valeur marchande.

De plus, IBM, qui dominait le marché, avait tout intérêt à permettre aux clients d'améliorer les programmes pour vendre plus d'ordinateurs, grâce à des logiciels adaptés<sup>2</sup> aux besoins d'une clientèle essentiellement constituée de techniciens de l'informatique.

Mais au cours des années 70, le *software* prend de l'importance et devient l'objet d'une véritable industrie, qui se destine désormais aux particuliers<sup>3</sup>. Les entreprises vont donc chercher sa réservation : c'est la fin de la liberté pour les utilisateurs et la fermeture des codes source.

2 - La protection du logiciel par un droit de propriété est apparue comme un choix évident, le droit commun étant trop lourd à mettre en œuvre<sup>4</sup>. Le logiciel étant une création immatérielle, c'est donc la propriété intellectuelle qui devait intervenir.

---

<sup>1</sup> Richard Stallman, « Le copyleft et son contexte », in *Copyright/Copywrong*, Actes du colloque Le Mans\ Nantes\ Saint-Nazaire\ février 2000, éditions Memo 2003, p. 216.

<sup>2</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *Les logiciels libres face au droit*, Namur , Facultés universitaires Notre-Dame de la paix, Bruxelles, Bruylant 2005, II.

<sup>3</sup> REVILLARD Diane, DIERMARK, *Livre blanc organisations et logiciels libres*, Version 1.0, 6 septembre 2005, p. 15.

<sup>4</sup> BITAN Hubert, *Droit des créations immatérielles*, Lamy, 2010, p. 19.

Mais une *summa divisio* au sein de la propriété intellectuelle existant entre propriété littéraire et artistique (droit d'auteur) et propriété industrielle (brevet et marque), il y eut hésitation entre une protection par brevet, du fait du caractère technique de la création informatique, et une protection par droit d'auteur.

Aux Etats-Unis, le Patent and Trademark Office étant hostile à la protection des logiciels par brevet, du fait de leur nature assimilable à un enchaînement d'étapes abstraites et/ou d'algorithmes mathématiques, c'est la protection par droit d'auteur qui a été retenue, position entérinée par les textes dès 1980<sup>5</sup>. De ce fait, les Etats-Unis, pour permettre aux logiciels américains d'être protégés de la même façon dans le reste du monde, ont souhaité que les autres Etats adoptent la même position. C'est ainsi que la Convention de Genève du 6 août 1952 imposa aux Etats signataires de protéger les logiciels par le droit d'auteur.<sup>6</sup>

En France, la question de la brevetabilité du logiciel a donc été écartée dès la loi du 2 janvier 1968<sup>7</sup>. Au niveau européen, la même position a été adoptée dans la Convention de Munich du 5 octobre 1973 relative à la délivrance des brevets européens. Restait alors la protection par le droit d'auteur. Mais la question suscita des controverses dans la jurisprudence française<sup>8</sup> et il fallut attendre l'intervention du législateur avec la loi du 3 juillet 1985 pour dissiper les doutes et faire entrer le logiciel parmi les œuvres de l'esprit protégeables par le droit d'auteur.

Devait suivre la directive européenne du 14 mai 1991, transposée en droit interne par la loi du 10 mai 1994, qui imposa aux Etats membres de protéger les programmes d'ordinateur par le droit d'auteur.

Reprise par l'article 10-1 de l'accord ADPIC et par l'article 4 du traité OMPI du 20 décembre 1996, la solution est devenue quasi universelle.

---

<sup>5</sup> Par une série d'amendement au Copyright Act de 1976.

<sup>6</sup> BITAN Hubert, *op. cit.*, p. 19.

<sup>7</sup> L'article L 611-10 2) c) du CPI le place effectivement dans la liste des exclusions des inventions.

<sup>8</sup> Un arrêt, notamment, jeta le trouble estimant que le logiciel ne pouvait constituer une œuvre de l'esprit. CA Paris, 13<sup>e</sup> ch., 4 juin 1984, JCP, E, 1985, II, 14409, note Vivant.

Remarquons que la question de la brevetabilité du logiciel ressurgit en Europe<sup>9</sup>. Suite à l'abandon du projet de directive européenne du 20 février 2002 proposant d'admettre la brevetabilité du logiciel, la présidente de l'Office Européen des Brevets a saisi, le 24 octobre 2008, la Grande Chambre des recours de l'OEB pour soumettre des questions relatives aux conditions d'admission d'un brevet portant sur un logiciel. Si cette saisine a été déclarée irrecevable par décision du 12 mai 2010<sup>10</sup>, la Grande Chambre des recours n'écarte pas cette protection du logiciel, mais renvoie à la jurisprudence des chambres de recours de l'OEB<sup>11</sup>. Pour le moment, s'il n'est toujours pas possible d'obtenir un brevet sur le programme « *en tant que tel* », car il ne possède pas de caractère technique<sup>12</sup>, la jurisprudence de l'OEB admet les brevets sur des inventions mises en œuvre par ordinateur<sup>13</sup>, dès lors que le programme produit un « *effet technique* ». La question du brevet logiciel reste donc d'actualité<sup>14</sup>.

En tout cas, la protection par droit d'auteur a été critiquée par les juristes eux-mêmes. Si le droit est sensé s'adapter aux évolutions de la société, il a été estimé que l'intégration du logiciel parmi les œuvres de l'esprit était inappropriée, malgré les ajustements de régime juridique opérés<sup>15</sup>.

C'est peut être aussi ce qui a conduit à l'émergence du mouvement libre. En effet, le droit d'auteur entraîne une durée de protection de 70 ans après la mort de l'auteur, ce

---

<sup>9</sup> Les offices de brevets des Etats-Unis ont admis, dès la décision de la Cour Suprême « *Diamond vs Diehr* » de 1981, des brevets sur des inventions contenant des programmes informatiques, puisque la protection par brevet n'avait pas été exclue par les textes.

<sup>10</sup> La Grande chambre des recours estime, en effet, qu'elle n'est pas une instance juridictionnelle en mesure de trancher de telles questions.

<sup>11</sup> « *Décision de la Grande Chambre des recours de l'OEB sur les brevets logiciels* », Site de l'APRIL, <http://www.april.org/fr/decison-de-la-grande-chambre-de-recours-de-loeb-sur-les-brevets-logiciels>.

<sup>12</sup> BITAN Hubert, *op. cit.*, p. 89.

<sup>13</sup> Par exemple : décision OEB « *Koch* » du 21 mai 1987 TOO26/86.

<sup>14</sup> Perspective qui ne satisfait pas non plus les défenseurs du logiciel libre dans la mesure où le brevet protégerait la fonctionnalité du logiciel, ce qui revient à protéger l'idée de créer un logiciel pour accomplir telle ou telle tâche. Or, les logiciels libres sont souvent des logiciels qui accomplissent des tâches similaires aux logiciels propriétaires. Le brevet logiciel pourrait ainsi bloquer le développement de ces logiciels.

Voir Site de l'organisation GNU, « *Sauver l'Europe des brevets sur les logiciels* », <http://www.gnu.org/philosophy/savingeurope.fr.html>.

<sup>15</sup> Ces critiques sont d'autant plus justifiées dans des systèmes tels que le droit d'auteur français, où l'on a une conception très personnaliste du droit d'auteur, conception difficilement transposable à des créations techniques.

qui semble démesuré au regard de la durée de vie d'un logiciel<sup>16</sup>. Surtout, le droit d'auteur a permis aux éditeurs de garder le secret autour du code source, méthode de fabrication du logiciel. Or, le logiciel étant une création technique nécessitant de la maintenance, cela peut paraître injustifié.

3 – A la fin des années 1970, le laboratoire d'intelligence artificielle du MIT<sup>17</sup> reçoit une nouvelle imprimante qui tombe régulièrement en panne. Richard Stallman, l'un des informaticiens, tente de récupérer le code source du logiciel qui pilote l'imprimante et de le modifier pour réparer les erreurs de conception, comme il était courant de le faire auparavant. Mais le code source du logiciel de l'imprimante n'est pas disponible et la société Xerox, propriétaire des droits d'auteur sur le logiciel, refusera de le lui communiquer.

Face à la croissance de cette nouvelle forme d'exploitation des logiciels, Richard Stallman en vient à la conclusion qu'il faut combattre ce qu'il désignera comme le logiciel « *propriétaire* »<sup>18</sup>.

En 1985, il quitte le MIT et crée une association à but non lucratif, la Free Software Foundation<sup>19</sup>, dédiée à la mise au point de logiciels librement copiables et modifiables.

Il décide alors de développer un système d'exploitation libre et ouvert baptisé GNU<sup>20</sup>. C'est la (re)naissance du logiciel libre. Cette décision est stratégique : le système d'exploitation constitue la pièce maîtresse d'un système informatique qui permet

---

<sup>16</sup> C'est d'ailleurs l'une des critiques formulées par Richard Stallman à l'égard des logiciels propriétaires. Voir MOUNIR Roderic, « Richard Stallman, un humaniste au cœur des logiciels », entrevue du 29 septembre 2009, disponible sur : <http://www.equilibres.ch/spip/spip.php?article237>.

<sup>17</sup> Massachusetts Institute of Technology, où travaillait Richard Stallman à l'époque.

<sup>18</sup> PERLINE et NOISSETTE Thierry, *La bataille du logiciel libre : dix clés pour comprendre*, La découverte 2004, p. 23.

<sup>19</sup> Site de la FSF, <http://fsffrance.org/>.

<sup>20</sup> Gnu's not Unix, car l'objectif premier de GNU est d'être similaire à Unix, système d'exploitation propriétaire, pour être rapidement utilisable, mais ce n'est pas Unix. Voir site de l'organisation GNU : <http://www.gnu.org/home.fr.html>.



d'accéder aux éléments de l'ordinateur et de faire tourner les programmes, c'est donc un préalable nécessaire au développement d'autres logiciels libres<sup>21</sup>.

En 1991, Linus Torvalds, un étudiant finlandais de l'université d'Helsinki, apporte la dernière touche à GNU et conçoit Linux, le noyau du système d'exploitation. Il ouvre ainsi la voie du système de travail coopératif qui va faire le succès de ces logiciels.

Une dizaine d'année après sa première mise à disposition, Linux est le concurrent numéro un de Microsoft.

4 – Le succès du logiciel libre est aujourd'hui incontestable. Ceci s'explique, tout d'abord, par le fait qu'il s'agit de concevoir des programmes de façon décentralisée, coopérative et ainsi, réaliser des tâches impossibles à accomplir de manière individuelle. L'accessibilité du code source par la communauté des utilisateurs facilite l'identification des erreurs et leur correction, ainsi que la mise en place des adaptations nécessaires. Il est ainsi possible de créer des logiciels d'une qualité au moins équivalente à celle des logiciels propriétaires.

Il faut également remarquer que le succès de l'internet n'est pas étranger au succès du mouvement libre<sup>22</sup>. Les réseaux ont assuré aux logiciels libres un moyen de diffusion rapide et peu onéreux, facilitant la communication à grande échelle entre les utilisateurs, ainsi que la gestion de projets de développement en commun<sup>23</sup>.

De plus, contrairement aux idées reçues, le logiciel libre s'inscrit parfaitement dans le système économique. En effet, s'est développée autour de ces logiciels, un peu particuliers, une industrie de services faisant prévaloir la vente de services sur la vente de produits. Ainsi, si les sociétés ou programmeurs, fournissent gratuitement,

---

<sup>21</sup> REVILLARD Diane, DIERMARK, *op. cit.*, p. 18.

<sup>22</sup> L'esprit du libre est né avec le Request For Comment (RFC), forme sous laquelle les spécifications techniques et les protocoles qui ont traités à Internet sont diffusés librement aux utilisateurs, de l'équipe qui a créé Internet en 1969.

<sup>23</sup> Jean Pierre ARCHAMBAULT, *L'industrie du logiciel à la croisée des chemins ? L'économie du libre*, Médialog, n°42 janv. 2002, p. 52.

ou pour des sommes modiques, un logiciel, elles vont, en revanche, vendre les services d'installation, de paramétrage, d'applications spécifiques... autour de ce logiciel<sup>24</sup>.

Enfin, au-delà d'un certain bénéfice économique direct, il y a également derrière les projets libres une volonté de reconnaissance. Que ce soit pour les entreprises ou pour les programmeurs, participer à l'élaboration d'un logiciel libre peut constituer un bon moyen de se faire connaître dans le monde informatique, et ainsi de trouver un emploi ou d'attirer des clients.

5 – Le succès est tel que le mouvement va faire des émules en dehors du monde informatique. Ainsi, en 2000, sous l'égide d'Antoine Moreau, la *copyleft attitude* a donné naissance à la LAL ou Licence Art Libre qui autorise à « *copier, diffuser, transformer librement les œuvres dans le respect des droits de l'auteur* ». <sup>25</sup>  
Vont suivre plusieurs types de licences dont notamment les licences Créative Commons<sup>26</sup> qui sont aujourd'hui celles qui connaissent le plus gros succès en matière d'art.

6 – Mais ce succès aurait pu ne pas être au rendez-vous.

La réalisation de logiciels libres n'aurait pas été possible sans la principale innovation de Richard Stallman : la licence GPL<sup>27</sup>, un contrat joint au logiciel libre qui garantit explicitement les libertés données aux utilisateurs.

En fait, la « bataille » entre logiciel libre et logiciel propriétaire se joue sur le terrain juridique, et plus précisément sur la protection par le droit d'auteur. C'est cette protection qui constitue le principal obstacle à la liberté des utilisateurs. Il fallait donc trouver un système permettant d'autoriser les utilisateurs à exécuter, copier,

---

<sup>24</sup> Idem, p.54.

<sup>25</sup> Site Artlibre, <http://artlibre.org/> .

<sup>26</sup> Site Créative Commons, <http://creativecommons.org/>.

<sup>27</sup> General Public Licence. Précisons que ce n'est pas la seule licence de logiciel libre, d'autres verrons le jour par la suite pour répondre à différents besoins.

distribuer et modifier librement le logiciel, alors que traditionnellement on utilise le droit d'auteur pour interdire cela. Ce système sera la licence.

Ainsi, un logiciel libre peut se définir comme un logiciel soumis à une licence qui autorise à l'utiliser, l'étudier, le modifier puis le redistribuer<sup>28</sup>.

Mais la licence GPL va plus loin puisqu'elle empêche l'appropriation du code source et de ses modifications. Cet effet de la licence a été nommé *copyleft*. Il est à l'origine des idées selon lesquelles les logiciels libres iraient à l'encontre du droit d'auteur et plus généralement de toute forme d'appropriation.

La licence GPL est ainsi très largement empreinte des idées libertaires de la Free Software Foundation, et plus précisément de Richard Stallman, qui a pour ambition la disparition des logiciels propriétaires. Or, ces idées, et l'intransigeance de certains défenseurs du libre<sup>29</sup>, ont compromis les chances de succès de ce mouvement dans le monde des affaires.

7 – C'est sur ce constat, qu'en 1998, va s'opérer une scission au sein du mouvement libre. A ce moment là, les investisseurs comprennent que le secteur du libre est en pleine expansion, mais sont plutôt retissant à investir dans un concept qu'ils jugent basé sur un mouvement de contestation politique, hostile à toute forme de commerce<sup>30</sup>. Pour une partie des militants du libre, il fallait donc se détacher des considérations éthiques et politiques de la Free Software Foundation pour convaincre le monde des affaires du potentiel des logiciels libres.

Les responsables de la distribution Debian<sup>31</sup>, principalement Bruce Perens<sup>32</sup> et Eric Steven Raymond<sup>33</sup>, vont alors instaurer une appellation plus séduisante pour le

---

<sup>28</sup> BERNAULT Carine et CLAVIER Jean-Pierre, *Dictionnaire de droit de la propriété intellectuelle*, Ellipses, 2008, p. 251.

<sup>29</sup> Richard Stallman se voit attribuer une image d'anarchiste. Voir PERLINE et NOISETTE Thierry, *op. cit.*, p. 12.

<sup>30</sup> REVILLARD Diane, DIERMARK, *op. cit.*, p. 20.

<sup>31</sup> Système d'exploitation exclusivement composé de logiciels libres dont GNU/Linux. Voir le site de l'organisation Debian, <http://www.debian.org/>.

<sup>32</sup> Rédacteur des *Free Software Guidelines* du projet Debian.

monde des affaires, l'*open source*, et vont créer l'Open Source Initiative (OSI), association dont le but est de promouvoir les logiciels open source. Ils décident, également, de mettre en place leur propre définition de ces logiciels, l'*Open Source Definition*.

L'Open Source Initiative va mettre l'accent sur les qualités de ce type de logiciel plutôt que sur la liberté, ce qui va susciter les critiques des militants attachés à l'idéologie véhiculée par la Free Software Foundation. Mais la stratégie employée par cette organisation a eu beaucoup de succès. Les logiciels open source ont investi les entreprises, et certaines se sont même spécialisées dans cette branche. Le terme *open source* est aujourd'hui sur toutes les lèvres, y compris celles des juristes.

8 – De nombreuses analyses des logiciels libres et open source, ainsi que des licences qui leur sont attachées, ont été effectuées. Les conditions de distribution de ces logiciels défiant quelque peu la logique du droit d'auteur, la doctrine s'est intéressée à la validité de ces licences et à l'intégration de ces logiciels dans le paysage juridique. Mais toutes ces analyses ont été faites de manière globale, sans réellement différencier les logiciels libres et les logiciels open source.

Il est vrai que, de prime abord, la différence paraît se situer sur des questions de méthode de promotion des logiciels, plutôt que sur des questions juridiques.

Toutefois, on constate que les termes *libre* et *open source* sont parfois utilisés pour désigner des logiciels qui ne sont pas du tout soumis au même régime juridique. On a pu, ainsi, voir les logiciels libres associés à des logiciels gratuits, ou encore les logiciels open source à des logiciels dont on peut simplement consulter le code source, le terme *open source* signifiant « à sources ouvertes ».

De même, au regard des objectifs des deux mouvements, on considère souvent que les logiciels libres sont soumis à des licences copyleftées qui assurent la pérennité de

---

<sup>33</sup> Développeur hors pair à l'origine de l'essai de référence sur la méthode de développement de ce type de logiciels : *la cathédrale et le bazar*.

la liberté, et que les logiciels open source sont soumis à des licences permissives. Enfin, on ne peut nier l'existence de ces deux mouvements et de leurs définitions concurrentes.

Or, la diffusion de tels logiciels ne pouvant plus, aujourd'hui, être considérée comme un « effet de mode », compte tenu de l'ampleur qu'elle a prise, il est essentiel de déterminer les contours de la notion de logiciel libre.

Comme l'existence de logiciels dits *open source* semble troubler ces contours, il convient d'étudier si et dans quelle mesure on peut distinguer un logiciel libre et un logiciel open source.

9 – Notre ambition sera ainsi de mettre fin aux amalgames suscités par la terminologie employée par ces deux mouvements ainsi que par leur idéologie, mais aussi de déterminer si la distinction libre/open source est juridiquement pertinente.

Dans une première partie, nous observerons que libre et open source ne constituent qu'un sous-ensemble de la distinction entre logiciel libre/open source et logiciel propriétaire et que, à ce titre, il existe une véritable analogie entre ces deux types de logiciels.

Puis, dans une seconde partie, nous analyserons dans quelle mesure les deux mouvements s'opposent. Ce qui nous amènera à constater que l'analogie entre logiciels libres et logiciels open source n'est pas parfaite.

# Partie 1 – Logiciel libre et open source : une analogie marquée.

10 – Il existe une analogie marquée entre logiciels libres et logiciels open source car, s'il y a deux mouvements, il s'agit d'une sorte de sous-division à la distinction plus large entre logiciel propriétaire et logiciel libre/open source. On parle souvent de ces deux mouvements comme d'une désignation concurrente pour un même type de logiciel. Ainsi, les deux termes tendent à recouvrir un concept commun (Chapitre 1).

Pour aboutir à ce résultat, nous verrons que les deux mouvements passent par un mécanisme contractuel : la licence. Ce mécanisme entraîne une certaine diversité, puisque plusieurs types de licence existent, en fonction des droits et obligations concédés. Cette diversité est également partagée par l'Open Source Initiative et la Free Software Foundation (Chapitre 2).

## **Chapitre 1 – Un concept commun**

11 – Un logiciel libre comme un logiciel open source ne se différencie pas des autres logiciels d'un point de vue technique ou fonctionnel<sup>34</sup>. Son originalité se trouve dans son régime juridique. C'est pour cela que nous parlons de concept de logiciel libre/open source, car il s'agit d'une nouvelle utilisation de la protection du logiciel et non pas d'un nouveau genre de logiciel. Ce concept consiste en une alternative au logiciel propriétaire et est commun au logiciel libre et au logiciel open source (Section I).

---

<sup>34</sup> Néanmoins, il est souvent affirmé qu'un logiciel libre ou open source a des qualités techniques supérieures à un logiciel propriétaire. Cela serait une conséquence du régime particulier auquel est soumis le logiciel, mais il n'y a aucune garantie sur ce point. Ce n'est pas l'objectif immédiat du concept.

Cette alternative est rendue possible par les licences, qui permettent de faire varier le régime juridique applicable au programme (Section II).

## **Section I – Une alternative au logiciel propriétaire**

12 – Nous l'avons vu, lorsque le logiciel a acquis une certaine valeur sur le marché informatique, les entreprises ont cherché sa réservation et, malgré le caractère technique des logiciels, le choix s'est porté sur le droit d'auteur faisant ainsi du logiciel une œuvre de l'esprit, bien que le régime juridique ait dû faire l'objet d'adaptations (A).

Comme nous l'avons fait remarquer, le logiciel libre/open source est souvent décrit comme allant à l'encontre du droit d'auteur, notamment en ce qu'il s'oppose au logiciel qualifié de propriétaire par Richard Stallman. Cependant, si nous verrons qu'il défie quelque peu la logique de la propriété littéraire et artistique, il faut garder à l'esprit que le logiciel libre/open source est avant tout un logiciel et, que, de ce fait, il reste soumis au droit d'auteur. Finalement, ce qui oppose le logiciel libre/open source au logiciel propriétaire est l'étendue des droits et obligations qui vont peser sur l'utilisateur (B).

### **A – La protection du logiciel par le droit d'auteur**

13 - Dans un premier temps, nous verrons ce qui est protégé par le droit d'auteur (1) puis nous étudierons son régime de protection (2).

## 1 – Les contours de la protection

14 – **Définition.** Il faut constater que ni la directive européenne du 14 mai 1991 relative à la protection des programmes d'ordinateur<sup>35</sup>, ni le Code de la Propriété Intellectuelle ne définissent ce qu'est un logiciel.

En droit français, pour savoir ce que recouvre la notion de logiciel, il faut alors se référer à l'arrêté du 22 décembre 1981 d'enrichissement de la langue française<sup>36</sup>. Ce texte définit le logiciel comme « *l'ensemble des programmes, procédés et règles, et éventuellement de la documentation, relatif au fonctionnement d'un ensemble de traitement de données* ». Reste à savoir comment cet ensemble peut donner prise à des droits de propriété intellectuelle.

15 – **Objet de la protection.** Le logiciel est envisagé par les textes comme une œuvre littéraire<sup>37</sup>, ce qui amène à se demander ce qui est protégé dans le logiciel. De prime abord, il faut signaler qu'aucune différence n'est faite entre logiciel d'exploitation (qui sert d'interface entre les logiciels d'application et le matériel informatique) et logiciel d'application, ni entre logiciel spécifique (logiciel conçu spécifiquement pour répondre à des besoins précis) et logiciel standard (ou progiciel).

En fait, ce n'est pas la fonctionnalité du logiciel qui va être protégée (c'est-à-dire « *la mise en œuvre de la capacité de celui-ci à effectuer une tâche précise ou à obtenir un résultat déterminé* »<sup>38</sup>), les idées n'étant pas protégées par le droit

---

<sup>35</sup> On peut noter que la loi française est une des seules à utiliser le terme « logiciel » et que la plupart des textes (directive de 1991, accord ADPIC et traité OMPI sur le droit d'auteur de 1996) emploie plutôt le terme « programme d'ordinateur ». Cependant, il semble que cette différence n'a pas une grande importance et que les deux notions se réfèrent à la même chose. LUCAS André, DEVEZE Jean, FRAYSSINET Jean, *Droit de l'informatique et de l'internet*, Thémis droit privé, Presses universitaires de France 2001, n°510 p. 302.

<sup>36</sup> JO 17 janvier 1982, p. 624.

<sup>37</sup> Exemple : « Les programmes d'ordinateur sont protégés en tant qu'œuvres littéraires... », Article 4 Traité OMPI.

<sup>38</sup> Cass. 1<sup>ère</sup> civ., 13 déc. 2005, n° 03-21.154.



d'auteur<sup>39</sup>, mais l'écriture informatique, l'enchaînement des instructions, c'est-à-dire la forme que prend l'idée.

La rédaction des instructions se fait en langage dit évolué, lisible par l'homme, ce qui représente le programme-source (ou code source), mais ce langage n'étant pas compréhensible par la machine, elles vont être compilées automatiquement en langage machine pour pouvoir être exécutées, donnant ainsi le programme-objet (ou code objet). L'article L 112-1 du Code de la Propriété Intellectuelle interdisant de tenir compte de la forme d'expression, la protection doit être indépendante du langage, et par conséquent le code source comme le code objet est protégé par le droit d'auteur.

L'article L 112-2 13° du Code de la Propriété Intellectuelle vise également « *le matériel de conception préparatoire* ». Concrètement, le législateur a voulu englober dans le champ de la protection « *l'ensemble des travaux de conception aboutissant au développement d'un programme, à condition qu'ils soient de nature à permettre la réalisation d'un programme d'ordinateur à un stade ultérieur* »<sup>40</sup>. Ainsi tous les documents qui permettent d'aboutir à la création du logiciel seront également protégés<sup>41</sup>.

En revanche, s'agissant de la documentation destinée à l'utilisateur, autrement dit le manuel d'utilisation, il semble qu'elle ne soit pas incluse dans la notion de logiciel. Néanmoins elle pourra faire l'objet d'une protection par le droit commun d'auteur si elle en présente les conditions<sup>42</sup>. La même solution sera retenue pour l'interface du logiciel.

---

<sup>39</sup> Le principe en droit de la propriété intellectuelle est que les idées sont de libre parcours et échappent à toute appropriation. Voir par exemple : Civ. 1<sup>ère</sup>, 25 mai 1992, RIDA oct. 1992, p.156. Sur la question des fonctionnalités voir : P. BELLOIR, « L'exclusion de la protection des fonctionnalités d'un logiciel par le droit d'auteur », RLDI, 2006, n°14.

<sup>40</sup> Considérant 7 de la directive communautaire 91/250 du 14 mai 1991.

<sup>41</sup> LUCAS André, DEVEZE Jean, FRAYSSINET Jean, *op. cit.*, n° 526, p. 317. Contra. CARON Christophe, *Manuel droit d'auteur et droits voisins*, 2<sup>e</sup> éd., Litec 2010, n° 179, p. 137, qui considère que le matériel de conception préparatoire contient le manuel d'utilisation et n'est pas protégé de la même façon que le logiciel.

<sup>42</sup> En ce sens : CA Paris, 4<sup>e</sup> ch., 1<sup>er</sup> juin 1994, Expertises, 1994, p.321.

## 2 – Le régime de la protection

16 – **Conditions de la protection.** En droit d'auteur, pour qu'une œuvre soit protégée deux conditions sont requises : il faut une création de forme et que cette forme soit originale. Le logiciel n'échappe pas à cette règle.

Pour ce qui est de la forme, nous l'avons dit, il s'agira de la rédaction des instructions. Dès lors que celle-ci aura commencé<sup>43</sup>, le logiciel pourra être protégé.

Concernant l'originalité, il faudra démontrer que les instructions présentent une certaine originalité. Mais le caractère technique du logiciel a posé problème car traditionnellement l'originalité se traduit par l'empreinte de la personnalité de l'auteur. La Cour de cassation a eu à traiter de cette question dans l'arrêt « Pachot »<sup>44</sup> et a estimé que l'originalité devait se traduire par la « *marque d'un apport intellectuel* »<sup>45</sup>. Cette solution a été quelque peu contestée car il y avait là une objectivisation du critère d'originalité. Pour autant, la directive européenne n'a pas apporté plus d'indications sur ce problème, puisqu'elle énonce que le logiciel sera protégé « *s'il est original, en ce sens qu'il est la création intellectuelle propre à son auteur* ».

Ainsi, dès les conditions de sa protection, le logiciel semble en marge des autres œuvres de l'esprit qui font l'objet d'une protection par le droit d'auteur. Cette situation se prolonge quant au régime de protection.

17 – **Les droits de l'auteur.** Comme pour toute œuvre de l'esprit, l'auteur d'un logiciel va se voir accorder sur sa création des droits patrimoniaux, qui permettent à l'auteur d'obtenir une rémunération contre l'utilisation de son œuvre, et des droits moraux, qui représentent l'expression du lien qui unit l'auteur à son œuvre et lui permettent d'en garder la maîtrise.

---

<sup>43</sup> L'article L 112-2 du CPI indique que la protection n'est pas subordonnée à l'achèvement de l'œuvre.

<sup>44</sup> Ass. Plen., 7 mars 1986, JCP, G, 1986, II, 20631, note Mousseron, Teyssié et Vivant.

<sup>45</sup> Il s'agira de démontrer que même si deux logiciels remplissent la même fonction, les codes qui permettent d'atteindre ce résultat ne sont pas les mêmes.

Concernant les droits patrimoniaux, l'auteur d'un logiciel dispose d'un droit de reproduction, d'un droit de traduction et d'adaptation, et d'un droit de distribution.

Tout d'abord, l'article L 122-6-1° du Code de la Propriété Intellectuelle permet à l'auteur d'autoriser ou d'interdire la « *reproduction permanente ou provisoire d'un logiciel en tout ou partie par tout moyen et sous toute forme* », et « *le chargement, l'affichage, l'exécution, la transmission ou le stockage de ce logiciel* » dans la mesure où ces actes « *nécessitent une reproduction* ».

Cette définition du droit de reproduction est dérogatoire au droit commun. En effet, elle est beaucoup plus large que celle du droit de reproduction accordé pour d'autres types d'œuvre, puisqu'en incluant la reproduction « *provisoire* », elle permet de soumettre au monopole de l'auteur une fixation transitoire dans la mémoire vive de l'ordinateur<sup>46</sup>.

Ensuite, l'article L 122-6-2° accorde à l'auteur un droit exclusif sur « *la traduction, l'adaptation, l'arrangement ou toute autre modification d'un logiciel et la reproduction du logiciel en résultant* ». Là encore, ce régime est dérogatoire au droit commun puisque, si la traduction, l'adaptation et l'arrangement sont habituellement rattachés au droit de reproduction et trouvent ainsi leur place au sein des droits patrimoniaux, il en va autrement de la « *modification* » qui normalement ne devrait être sanctionnée que si elle porte atteinte à l'intégrité de l'œuvre (soit en vertu du droit moral). Une fois de plus, le monopole de l'auteur a été très largement entendu<sup>47</sup>.

Enfin, l'article L 122-6-3° réserve à l'auteur « *la mise sur le marché à titre onéreux ou gratuit, y compris la location, du ou des exemplaires d'un logiciel par tout procédé* »<sup>48</sup>.

---

<sup>46</sup> Cela permet également à l'auteur de s'opposer à l'installation du logiciel sur plusieurs postes, même si ceux-ci appartiennent à l'utilisateur légitime. En ce sens : A. Strowel et J-P Triaille, *Le droit d'auteur, du logiciel au multimédia*, Story-Scientia et Bruylant, 1997, n° 250 ; et CLEMENT-FONTAINE Mélanie, *La licence publique générale GNU*, mémoire de DEA sous la direction de Monsieur le Professeur Michel VIVANT, Montpellier I/ CNRS 1999, p. 28.

<sup>47</sup> LUCAS André, DEVEZE Jean, FRAYSSINET Jean, *op. cit.* n°544 et s.

<sup>48</sup> Ce droit est cependant soumis au principe de l'épuisement des droits, c'est-à-dire qu'il ne pourra plus être exercé après la première vente d'un exemplaire, par l'auteur ou avec son consentement, dans l'un des pays membres de la Communauté européenne ou de l'EEE (art. L 122-6-3°).

Concernant les droits moraux, il faut tout d'abord signaler que la loi de 1985 a purement et simplement exclu le droit de repentir qui permet à l'auteur, qui a cédé tout ou partie de ses droits patrimoniaux, de revenir sur son engagement.

Autre dérogation au droit commun, le droit au respect de l'œuvre est beaucoup plus limité que pour les autres œuvres de l'esprit, puisque l'auteur ne peut « *s'opposer à la modification du logiciel par le cessionnaire des droits [...] lorsqu'elle n'est préjudiciable ni à son honneur ni à sa réputation* »<sup>49</sup>. Ainsi, ce droit, en matière de logiciel, paraît symbolique tant il est rare qu'une modification du logiciel porte atteinte à l'honneur ou à la réputation de son auteur.

Parallèlement, le Code de la Propriété Intellectuelle ne fait pas référence au droit à la paternité pour l'auteur d'un logiciel. Pour autant, tout le monde s'accorde à dire que ce droit existe par application du droit commun d'auteur, l'article L 121-7 sur les droits moraux de l'auteur d'un logiciel n'étant qu'une dérogation à ce droit commun<sup>50</sup>. Reste la question du droit de divulgation. Si l'on applique le même raisonnement que pour le droit à la paternité, il faut considérer que l'auteur d'un logiciel en bénéficie. Mais ce droit, qui permet à l'auteur de maîtriser la communication de son œuvre au public et, dans ce sens, lui offre la possibilité de revenir sur un éventuel engagement auprès d'un cessionnaire<sup>51</sup>, pose problème d'un point de vue économique en matière de logiciel<sup>52</sup>. Dans la mesure où la question n'a pas été tranchée, nous considérons qu'il faut prendre en compte ce droit de divulgation.

Ainsi, la législation confère une large gamme de droits à l'auteur d'un logiciel pour protéger sa création et en tirer profit. Nous allons voir qu'il n'en va pas différemment pour un logiciel dit libre ou open source.

---

<sup>49</sup> Article L 121-7-1° du CPI.

<sup>50</sup> LUCAS André, DEVEZE Jean, FRAYSSINET Jean, *op. cit.* n° 556, p. 339.

<sup>51</sup> En ce sens : Cass. 14 mars 1900, « Eden c/ Whistler », D.P. 1900, 1, 497, rapport Rau, concl. Desjardins, note Planiol ; S. 1900, 1, p. 489.

<sup>52</sup> En ce sens : VIVA NT Michel, « Logiciel 94 : tout un programme ? », JCP, G, 1994, I, 3792, note 30.

## B – Logiciel propriétaire versus logiciel libre/open source

18 – Logiciels libres et logiciels open source, sont des œuvres de l'esprit soumis au droit d'auteur, au même titre qu'un logiciel propriétaire (1). Néanmoins, l'étendue des droits dont dispose l'utilisateur sur un logiciel libre/open source et sur un logiciel propriétaire diffère (2).

### 1 – Des logiciels soumis au droit d'auteur

19 – **Un logiciel avant tout.** Un logiciel libre comme un logiciel open source est avant tout un logiciel classique, qui peut être un logiciel d'exploitation ou d'application, un logiciel spécifique ou un progiciel.

Or, l'article L 111-1 du Code de la Propriété Intellectuelle énonce que « *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* ». Ainsi, le droit d'auteur sur une œuvre né automatiquement sans qu'il soit nécessaire d'effectuer un dépôt ou un enregistrement<sup>53</sup>.

Dès lors, l'auteur d'un logiciel, si celui-ci répond aux conditions de la protection, c'est-à-dire s'il s'agit d'une création de forme originale, bénéficie obligatoirement sur cette création des droits que nous avons déjà énoncés.

20 – **Des logiciels appropriés.** On voit alors que le fait d'opposer le logiciel libre/open source au logiciel « propriétaire » est assez trompeur, car en fait il s'agit également d'un logiciel approprié : il appartient à son auteur qui bénéficie sur celui-ci de droits exclusifs<sup>54</sup>. Comme le fait remarquer Benoît de Roquefeuil, « *conceptuellement le logiciel libre au sens de l'œuvre de tout le monde (donc de*

---

<sup>53</sup> Contrairement aux créations protégées par le droit de la propriété industrielle, telles que les marques ou les inventions.

<sup>54</sup> Voir notamment : CARON Christophe, « Les licences de logiciels dits « libre » à l'épreuve du droit d'auteur français », Dalloz 2003, chron. p. 1556.

*personne) est un non-sens juridique* »<sup>55</sup>. Cette affirmation vaut également pour un logiciel open source.

Pourtant, il existe bien une différence entre ces deux types de logiciel.

## 2 – Une étendue des droits différente

21 – **Le logiciel libre.** On définit le logiciel libre en se référant à la définition donnée par la Free Software Foundation<sup>56</sup>. Celle-ci a déterminé quatre libertés que l'on doit retrouver dans un logiciel pour que l'on puisse considérer celui-ci comme libre :

- la liberté d'exécuter le programme, pour tous les usages (liberté 0)
- la liberté d'étudier le fonctionnement du programme et de l'adapter à ses besoins (liberté 1)
- la liberté de redistribuer des copies (liberté 2)
- la liberté d'améliorer le programme et de publier ces améliorations (liberté 3)

Il est également précisé que pour tout cela l'accès au code source du programme est une condition requise.

22 – **Le logiciel propriétaire.** En fait, le terme de logiciel propriétaire est apparu afin de nommer le logiciel qui n'est pas un logiciel libre et se définit donc par opposition à ce dernier<sup>57</sup>. Il est utilisé pour désigner tout logiciel dont l'auteur se réserve l'exclusivité de l'exploitation.

23 – **La distinction.** Tout d'abord, un logiciel libre doit être libre d'utilisation. Lorsque l'on acquiert un logiciel propriétaire, on bénéficie généralement de cette

---

<sup>55</sup> De ROQUEFEUIL Benoît, « Le statut juridique des logiciels libres », Gaz. Pal., janv. 2007, n°17, p. 10.

<sup>56</sup> « Définition d'un logiciel libre », Site de l'organisation GNU,  
<http://www.gnu.org/philosophy/free-sw.fr.html>.

<sup>57</sup> Le terme logiciel propriétaire est également employé pour lever l'ambiguïté avec la notion de logiciel commercial.

liberté d'utilisation<sup>58</sup>. Cependant, le degré de liberté n'est pas identique, car conformément à la définition du logiciel libre, rien ne doit faire obstacle à cette libre utilisation, exigence que l'on ne retrouve pas dans un logiciel propriétaire. Par exemple, grâce au droit de reproduction, il est possible d'interdire l'installation du programme sur plusieurs postes (même s'ils appartiennent tous à l'utilisateur légitime). Cela n'est pas possible avec un logiciel libre.

Un logiciel libre doit également être librement modifiable. Cela n'est jamais le cas pour un logiciel propriétaire. Ce point a d'ailleurs posé problème car un programme est avant tout un outil qui peut faire l'objet de dysfonctionnements et refuser la modification du logiciel, c'est refuser la possibilité à l'utilisateur de corriger des erreurs qui rendent le logiciel impropre à l'usage pour lequel il est destiné. C'est pourquoi le législateur a prévu une exception d'analyse<sup>59</sup>, mais rien de comparable à ce que permet un logiciel libre.

L'utilisateur d'un logiciel libre doit aussi être en mesure d'effectuer librement des copies du logiciel. Là encore, cela n'est pas permis avec un logiciel propriétaire malgré l'exception de copie de sauvegarde<sup>60</sup> prévue par le législateur. En effet, cette exception reste limitée, puisqu'une seule copie est autorisée et qu'elle ne peut être utilisée qu'en cas de problème avec l'utilisation du logiciel. Avec un logiciel libre, on peut effectuer autant de copies que l'on souhaite, peu importe leur destination (y compris s'il s'agit de les vendre).

Enfin, l'utilisateur d'un logiciel libre peut redistribuer le programme avec ou sans modifications, gratuitement ou non. On ne retrouvera pratiquement jamais une

---

<sup>58</sup> Ces logiciels sont effectivement commercialisés par le biais de contrats de licence utilisateur final (CLUF) qui confère un droit d'usage à l'utilisateur. Lorsque l'on « achète » un logiciel, il n'y a généralement pas transfert de propriété sur les droits d'auteur.

<sup>59</sup> Article L 122-6-1 III du CPI : « la personne ayant le droit d'utiliser le logiciel peut sans l'autorisation de l'auteur observer, étudier ou tester le fonctionnement de ce logiciel afin de déterminer les idées et principes qui sont à la base de n'importe quel élément du logiciel lorsqu'elle effectue toute opération de chargement, d'affichage, d'exécution, de transmission ou de stockage du logiciel qu'elle est en droit d'effectuer ».

<sup>60</sup> Article L 122-6-1 I du CPI : « la personne ayant le droit d'utiliser le logiciel peut faire une copie de sauvegarde lorsque celle-ci est nécessaire pour préserver l'utilisation du logiciel ».

telle possibilité avec un logiciel propriétaire car ce sont des programmes essentiellement destinés à la commercialisation<sup>61</sup>.

Il faut également préciser que le code source d'un logiciel propriétaire est le plus souvent gardé secret de façon à rendre plus difficile une atteinte aux droits de l'auteur, alors qu'un tel secret n'a plus d'intérêt avec un logiciel libre et irait même à l'encontre des libertés octroyées à l'utilisateur.

**24 - Le logiciel open source.** Le logiciel open source, quant à lui, se définit en référence à l'Open Source Definition<sup>62</sup> (OSD) donnée par l'Open Source Initiative. Le logiciel est open source si la licence qui lui est applicable répond à dix critères :

- elle prévoit la libre redistribution du logiciel (le paiement de royalties ne peut être exigé),
- elle prévoit la mise à disposition du code source,
- elle prévoit la possibilité de réaliser des travaux dérivés et de les redistribuer,
- elle prévoit la préservation de l'intégrité du code source (un tiers ne peut pas s'approprier le travail d'un autre et les contributions de chacun sont clairement attribuées),
- elle n'effectue pas de discrimination entre les personnes et les groupes (les droits accordés peuvent bénéficier à n'importe qui),
- elle n'effectue pas de discrimination entre les domaines d'application (la licence ne doit pas limiter le champ d'application du programme),
- elle s'applique sans dépendre d'autres contrats (par exemple on ne peut imposer d'accord de non divulgation),
- elle ne doit pas être propre à un produit (elle est attachée au code, non à un logiciel particulier),

---

<sup>61</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°212, p. 138. Cependant, on peut trouver des logiciels propriétaires que l'on peut redistribuer, tels que les freewares.

<sup>62</sup> « The Open Source Definition », Site de l'Open Source Initiative, <http://www.opensource.org/docs/osd>.



- elle ne doit pas contaminer d'autres logiciels distribués avec le programme qu'elle couvre,
- elle doit être technologiquement neutre.

On remarque que, comme le logiciel libre, un logiciel open source peut être librement redistribué, modifié et son code source est mis à disposition des utilisateurs.

Bien que n'étant pas expressément prévue par la définition, on va également retrouver la liberté de copier ce logiciel. En effet, on peut estimer qu'elle découle de la liberté de redistribuer le logiciel, car si on ne peut effectuer de copie, la possibilité de distribuer le logiciel en sera automatiquement affectée<sup>63</sup>. D'ailleurs, on sait qu'en droit français, le droit de distribution constitue le prolongement du droit de reproduction, ce qui atteste de leur lien.

Quant à la liberté d'utiliser le logiciel, nous verrons que ce n'est pas la priorité de l'Open Source Initiative, qui s'intéresse plutôt à la contribution. Mais, d'une part, comment permettre la modification sans permettre l'utilisation, d'autre part, l'*Open Source Definition* contient un critère de non discrimination entre les domaines d'application, ce qui rejoint la libre exécution pour tout type d'usage<sup>64</sup> exigée pour le logiciel libre.

Ainsi, on voit que le logiciel open source comme le logiciel libre est un logiciel soumis au droit d'auteur mais pour lequel, contrairement au logiciel propriétaire, l'auteur ne se réserve pas la plupart des actes d'exploitation<sup>65</sup>. Nous allons voir que, si cela constitue une pratique nouvelle, ce n'en n'est pas moins une pratique permise par la législation.

---

<sup>63</sup> De plus, on remarque que la définition de la FSF parle de « liberté de redistribuer des copies », ce qui montre l'imbrication des deux libertés dans l'esprit de la communauté libre/open source.

<sup>64</sup> La définition énonce effectivement que « *par exemple, elle (la licence) ne doit pas interdire l'utilisation du programme pour faire des affaires ou dans le cadre de la recherche génétique* ».

<sup>65</sup> Il faut également remarquer que l'OSD est issue du Contrat Social Debian de GNU/Linux, et permettant de classer les logiciels en deux catégories : libre ou non. Or, ce contrat a été approuvé par la FSF, ce qui confirme que libre et open source reposent sur les mêmes bases. Voir PERENS Bruce, « La définition de l'Open Source », extrait de *Open Sources - Voices from the Open Source Revolution*, 1999, ISBN 1-56592-582-3, disponible sur : [http://www.linux-france.org/article/these/the\\_osd/fr-the\\_open\\_source\\_definition.html#toc2](http://www.linux-france.org/article/these/the_osd/fr-the_open_source_definition.html#toc2).

## **Section II – Une possibilité réalisée par le mécanisme contractuel**

25 - Pour obtenir un logiciel libre/open source, il faut passer par le mécanisme contractuel (A). Néanmoins, ce mécanisme soulève quelques difficultés juridiques, cela que l'on soit en présence d'une licence libre ou d'une licence open source (B).

### **A – La licence comme instrument de permission**

26 – Le concept de logiciel libre/ open source consiste en fait pour l'auteur à utiliser une faculté que lui confère la législation : celle d'autoriser les actes d'exploitation sur son œuvre (1). L'exercice de cette faculté nécessite, cependant, une manifestation de volonté (2).

#### **1 – La faculté d'autorisation**

27 – **Durée de protection du logiciel.** Comme nous l'avons dit, le logiciel, qu'il soit libre/open source ou non, est protégé par le droit d'auteur dès sa création. La loi du 3 juillet 1985 limitait la durée des droits patrimoniaux à 25 ans à compter de la date de cette création. Cette solution, qui prenait des libertés avec la Convention de Berne<sup>66</sup>, témoignait de la spécificité du logiciel. Mais elle a été abandonnée par la loi du 10 mai 1994, et, désormais, la durée de protection du logiciel est équivalente à celle des autres œuvres de l'esprit, c'est à dire la vie de l'auteur et 70 ans après sa mort.

Dès lors, tant que ce délai n'est pas écoulé l'auteur du logiciel bénéficie du monopole d'exploitation sur sa création et toute atteinte à ce monopole constitue un acte de contrefaçon. En effet, le droit d'auteur a été construit pour protéger les œuvres

---

<sup>66</sup> L'article 7 de la Convention prévoit une durée de protection, notamment pour les œuvres littéraires, qui est la vie de l'auteur et 50 ans après sa mort.

de l'esprit et permettre à leurs auteurs d'en retirer un revenu afin de les inciter à la création, considérée comme moteur de développement de la société.

L'objectif est donc de conférer à l'auteur la maîtrise de son œuvre par le biais d'un droit exclusif, « *arme la plus efficace au service des auteurs qu'elle place en position de force* » selon André et Henri-Jacques Lucas<sup>67</sup>.

**28 – Maîtrise des droits.** Mais, ce droit exclusif se manifeste par un droit, pour l'auteur, d'autoriser ou d'interdire un certain nombre d'actes d'exploitation qui correspondent à ses droits patrimoniaux, puisque, à la différence des droits moraux, ceux-ci sont à la disposition des auteurs.

Si le principe reste, dans le silence de l'auteur, l'interdiction, rien n'interdit à l'auteur d'autoriser tous ces actes, de « libérer » les droits d'exploitation. C'est précisément sur ce point que se situe la différence entre un logiciel dit libre ou open source et un logiciel dit propriétaire : alors qu'avec un logiciel propriétaire la plupart des droits d'exploitation seront réservés à l'auteur, avec un logiciel libre/ open source tous les droits sont « libérés ».

Toutefois, l'abstention ou le silence de l'auteur ne présupposant pas une liberté d'utilisation<sup>68</sup>, seule une manifestation de volonté de la part de l'auteur peut permettre une telle libération.

## 2 – Manifestation de volonté par le contrat

**29 – Domaine public consenti.** La possibilité de faire tomber précocement l'œuvre dans le domaine public, pour « libérer » le logiciel, par un acte de renonciation, a été envisagée par Mélanie Clément-Fontaine<sup>69</sup> : il s'agirait alors d'un domaine public consenti qui viendrait s'ajouter au domaine public traditionnel. Il existe une doctrine, datant du début du XXe siècle, qui a développé une théorie

---

<sup>67</sup> LUCAS A. et H-J., *Traité de la propriété littéraire et artistique*, 3<sup>e</sup> éd., Litec, 2006.

<sup>68</sup> De ROQUEFEUIL Benoît, op. cit., n°17, p. 11.

<sup>69</sup> Cette théorie est parfaitement exposée dans sa thèse « Les œuvres libres ». Voir CLEMENT-FONTAINE Mélanie, *Les œuvres libres*, thèse de doctorat, Montpellier I, 2006.

générale de la renonciation, théorie qui pourrait s'appliquer, selon M. Clément-Fontaine, en matière de logiciel libre. La renonciation consisterait en un acte de volonté, qui ne se présume pas. Cet acte serait irrévocable et beaucoup plus efficace que le simple fait de ne pas exercer ses droits, car la renonciation serait opposable à tous.

Cependant, ni le Code de la propriété intellectuelle, qui pourtant reconnaît cette faculté au titulaire d'un brevet<sup>70</sup> et au titulaire d'une marque<sup>71</sup>, ni la jurisprudence ne semble admettre cette possibilité. L'argument le plus fort, s'opposant à une telle possibilité, est peut être celui du parallélisme des formes : les droits sur l'œuvre naissent et s'éteignent par l'effet de la loi, par conséquent, il semble exclu que l'auteur puissent y renoncer du seul fait de sa volonté. De plus, le droit d'auteur français étant construit sur une logique personnaliste, le législateur a privé l'auteur de la possibilité de renoncer à ses droits moraux. De ce fait, un véritable acte de renonciation ne pourrait avoir d'effet qu'à l'égard des droits patrimoniaux.

**30 – La licence.** Que ce soit pour rendre un logiciel libre ou open source, il va donc falloir passer par un autre mécanisme : la licence.

Toutefois, ce qui semble être un passage obligé en droit français, nous semble pouvoir être considéré comme un choix pour ceux qui ont initié le concept de logiciel libre. En fait, il semble que le copyright, aux Etats-Unis<sup>72</sup>, permet à l'auteur de renoncer à ses droits et, ainsi, de mettre volontairement une œuvre dans le domaine

---

<sup>70</sup> Article L 613-24 du CPI.

<sup>71</sup> Article L 714-2 du CPI.

<sup>72</sup> « Rights gained under the Copyright Law, 17 U.S.C.A. § 1 et seq., may be abandoned. Abandonment of such rights, however, must be manifested by some overt act indicative of a purpose to surrender the rights and allow the public to copy. », Juge Frederick Hamley, dans la décision « Hampton v. Paramount Pictures Corporation », United States Court of Appeals Ninth Circuit, 2 juin 1960, disponible sur : <http://openjurist.org/279/f2d/100#p19>.

Cette possibilité est également admise dans l'affaire "Schatt v. Curtis Management Group Inc.", 764 F Supp. 902 907, SDNY 1991, [http://scholar.google.com/scholar\\_case?case=11610835120635768892&q=schatt+v+curtis&hl=en&as\\_sdt=100000002](http://scholar.google.com/scholar_case?case=11610835120635768892&q=schatt+v+curtis&hl=en&as_sdt=100000002).

Avant 1989, un tel abandon pouvait être déduit de l'absence de notice de copyright suivant la théorie de l'*estoppel*, mais depuis la notice n'est plus nécessaire pour prouver la protection de l'œuvre et il faudrait donc un acte manifestant expressément l'intention de l'auteur de placer son œuvre dans le domaine public.

public. Mais, grâce à la licence, l'auteur va pouvoir définir les conditions d'utilisation, de modification et de partage de son œuvre, ce qui présente quelques avantages que l'on ne retrouve pas en cas de renonciation aux droits<sup>73</sup>.

Tout d'abord, l'auteur ne renonçant pas à ses droits, il lui sera toujours possible de les invoquer en cas de violation de la licence, ce qui permet d'assurer que sa volonté de partage soit respectée. D'ailleurs, l'intérêt est double puisqu'il sera alors possible non seulement de se fonder sur la contrefaçon, mais aussi sur la responsabilité contractuelle<sup>74</sup>.

Ensuite, cela lui permet également de mettre en place des dispositions autres que celles relatives au droit d'auteur. On pense ici à des dispositions concernant les brevets, les marques ou encore les DRM (Digital Rights Management)<sup>75</sup>. De telles dispositions ne sont pas négligeables, notamment celles concernant les brevets puisqu'elles permettent d'éviter un effondrement du système dans le cas où l'un des auteurs, originaires ou contributeurs, viendrait à déposer un brevet sur le logiciel. Cela n'est évidemment pas possible avec un acte de renonciation qui ne lie pas les tiers, mais leur est simplement opposable.

Enfin, l'auteur n'est pas animé par une intention libérale lorsqu'il consent une telle liberté, il va chercher à en retirer quelque chose, principalement, une certaine notoriété. Or, la licence lui permet cela, puisqu'il va pouvoir déterminer les conditions dans lesquelles il souhaite que son droit de paternité s'applique<sup>76</sup>.

---

<sup>73</sup> En ce sens : COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°33, p. 21.

<sup>74</sup> La décision, récente, « Jacobsen vs Katzer » aux Etats-Unis, a d'ailleurs affirmé cette possibilité de faire exécuter les termes de la licence en vertu du droit d'auteur et pas uniquement du droit des contrats, ce qui n'est pas sans conséquence sur l'éventuelle indemnisation de l'auteur. Voir US Court of Appeals for the Federal District, « Jacobsen vs Katzer », 13 août 2008, [http://scholar.google.com/scholar\\_case?case=17776182574171214893&q=535+F.3d+1373&hl=en&as\\_sdt=100000002](http://scholar.google.com/scholar_case?case=17776182574171214893&q=535+F.3d+1373&hl=en&as_sdt=100000002).

<sup>75</sup> Par exemple, l'Open Software License (OSL) v. 3.0, dans son article 10, contient une clause sur la résolution immédiate du contrat pour le cas où le licencié intenterait une action contre le donneur de licence sur la base de la violation d'un brevet que ce dernier aurait commise. Voir <http://www.opensource.org/licenses/osl-3.0.php>.

<sup>76</sup> Pour plus de détail, voir infra n° 93.

Mais la plus connue des conditions est peut être celle du *copyleft*, que nous verrons plus avant, et qui montre indubitablement tout l'intérêt de passer par un tel mécanisme<sup>77</sup>.

L'abandon des droits par l'auteur n'est donc pas, selon nous, la meilleure solution, ni pour l'Open Source Initiative, qui cherche à convaincre les entreprises de développer ce type de logiciels (on voit mal une société abandonner tout droit sur son logiciel), ni pour la Free Software Foundation, qui souhaite assurer la pérennité de la liberté accordée. C'est donc bien par un mécanisme contractuel, la licence, que l'on va pouvoir obtenir un logiciel libre comme un logiciel open source.

On peut remarquer que certains ont comparé les licences libres/open source les plus permissives à des actes de renonciation à ses droits par l'auteur, qui avaient pour effet de faire tomber les logiciels dans le domaine public. Mais cette licence n'a pas pour objet de faire disparaître ces droits, mais bien au contraire, de les exercer, même s'il s'agit d'autoriser plutôt que d'interdire. Par conséquent, il est inexact de parler de licence de domaine public.

De plus, comme nous le verrons<sup>78</sup>, si les logiciels tombés dans le domaine public répondent aux critères posés par la Free Software Foundation<sup>79</sup>, cette définition n'a pas été conçue pour ces logiciels, mais bien pour ceux qui sont encore soumis au droit d'auteur. De ce fait, lorsque l'on parle de logiciel libre, on désigne un logiciel soumis à une licence comme pour un logiciel open source.

Enfin, la théorie, selon laquelle cette licence ne constituerait pas un contrat, a été émise, notamment dans le but de faire échapper ce type de licence au droit des

---

<sup>77</sup> A savoir, qu'en cas d'abandon des droits, rien ne pourra empêcher quiconque de modifier l'œuvre et de soumettre l'œuvre ainsi modifiée à une licence propriétaire. Richard Stallman le dit lui-même : « La raison pour laquelle j'ai utilisé le droit d'auteur pour vous rendre ces libertés essentielles est pour imposer une condition : quand vous diffusez des copies, vous devez le faire avec cette même licence. », Richard Stallman, « Le copyleft et son contexte », op. cit., p. 217.

Mélanie Clément-Fontaine le remarque aussi par la suite. Voir : CLEMENT-FONTAINE Mélanie, « Faut-il consacrer un statut légal de l'œuvre libre ? », Propriétés Intellectuelles, janv. 2008, n°26, p. 75.

<sup>78</sup> Voir infra n° 96.

<sup>79</sup> A condition que le code source du logiciel soit matériellement mis à disposition, ce qui n'est pas toujours le cas.

contrats<sup>80</sup>. Cependant, cette théorie doit être rejetée en droit français. En effet, la seule autre qualification envisageable serait celle d'engagement unilatéral de volonté. Or, la majorité des auteurs français n'est pas favorable à la reconnaissance d'un tel acte, car ce serait un non sens que d'admettre une obligation sans créancier<sup>81</sup>. Mais, surtout, dans toute licence libre ou open source, il existe des obligations à charge de l'utilisateur : au minimum, on retrouvera une obligation de conserver certaines mentions relatives à l'exclusion de garantie et à l'identification de l'auteur<sup>82</sup>. Seul le contrat peut expliquer cela. De fait, ces licences se voient soumises au droit commun des contrats comme au droit d'auteur.

Cependant, la réception des licences libres et open source en France a suscité des interrogations.

## **B – Un instrument valable ?**

31 – Qu'il s'agisse des licences libres ou des licences open source, ces contrats, qui constituent une utilisation particulière du droit d'auteur, ont suscité l'attention de la doctrine quant à leur validité (1), mais, malgré la persistance de certains doutes, la jurisprudence en respecte les termes (2).

### **1 - Les interrogations sur la validité**

32 – **Qualification de la licence.** Logiciel libre et logiciel open source correspondent à un concept commun, qui est de rendre possible la libre utilisation, copie, distribution et modification du logiciel. L'opération réalisée par les licences, qui permettent d'obtenir ce résultat, est donc identique, ce qui nous permet de dire qu'elles peuvent faire l'objet d'une qualification unique. Cette question de la

---

<sup>80</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°34, p. 21.

<sup>81</sup> LARROUMET Christian, *Droit civil- Les obligations Le contrat*, Tome III, 5<sup>e</sup> éd., Economica 2003, p.75.

<sup>82</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°35, p. 22.

qualification des licences libres/open source a reçu des réponses différentes de la part de la doctrine.

Comme nous le verrons par la suite<sup>83</sup>, il convient d'effectuer une dissociation entre l'opération qui a lieu sur le logiciel, en tant qu'outil immatériel, et celle qui est relative aux droits d'auteur attachés au logiciel, en tant qu'œuvre de l'esprit. Il nous semble alors que la prestation caractéristique de ces licences consiste en la mise à disposition des droits patrimoniaux, ce qui nous permet de qualifier ces licences de contrats gracieux, car il n'y a pas de contrepartie financière pour cette mise à disposition des droits d'auteur. Sur ce point nous renvoyons aux développements ultérieurs<sup>84</sup>. De ce fait, il semble que pour l'ensemble des auteurs, les qualifications de vente et de louage d'ouvrage soient à écarter<sup>85</sup>. Pour la même raison, on peut exclure la qualification de louage de biens incorporels<sup>86</sup>, prévue par l'article 1709 du Code civil, déjà envisagée pour les licences de propriété industrielle.

Reste la qualification de commodat, ou prêt à usage, mais le contrat doit être conclu pour un « *certain temps* ». Or, les licences libres/open source ne prévoient jamais une durée à l'issue de laquelle une restitution aurait lieu. Surtout une telle restitution se conçoit mal en matière de logiciel<sup>87</sup>. Nous rejoignons alors une partie de la doctrine qui estime qu'il n'est pas nécessaire de contorsionner les catégories de contrats définis par la législation, puisqu'il suffit de recourir à celle de contrat innomé du droit civil.

Au regard du droit d'auteur, ces contrats ne se retrouvent pas non plus parmi les figures envisagées par la législation. Mais la difficulté vient d'ailleurs. Les licences libres comme open source n'emportent pas transfert du droit de propriété intellectuelle, il n'y a pas cession pure et simple puisque l'accès à l'œuvre n'est pas

---

<sup>83</sup> Voir infra n° 69.

<sup>84</sup> Voir infra n° 68.

<sup>85</sup> Voir COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°220 et s., ou encore CLEMENT-FONTAINE Mélanie, *La licence publique générale GNU*, *op. cit.*, p. 12-13.

<sup>86</sup> Envisagée par Mélanie Clément-Fontaine, *idem*, p. 14.

<sup>87</sup> Si l'on peut concevoir la restitution du support, celle du logiciel dématérialisé paraît difficile et n'aurait que peu de sens puisque le donneur de licence ne se dessaisit pas du logiciel.

Voir COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°227, p. 148, ou encore Conseil supérieur de la propriété littéraire et artistique, *La mise à disposition ouverte des œuvres de l'esprit*, juin 2007, p. 23, disponible sur : <http://www.cspla.culture.gouv.fr/CONTENU/miseadiposouverterapp.pdf>.



réservé au bénéficiaire de la licence<sup>88</sup>. Ce qui, comme l'explique Mélanie Clément-Fontaine<sup>89</sup>, rend l'appellation de licence, qui signifie autorisation, particulièrement pertinente. Mais, en droit français, si la distinction entre cession et licence est classique en matière de propriété industrielle, elle n'existe pas en droit d'auteur : seule la cession est envisagée.

Pour autant, il semble que la seule mise à disposition des droits ne soit pas exclue par la législation. C'est pourquoi, certains auteurs ont pu considérer que le terme « *cession* », employé par le Code de la propriété intellectuelle, devait être entendu au sens large, comme comprenant les contrats d'exploitation visés par le terme « licence »<sup>90</sup>. En fait, pour certains auteurs<sup>91</sup>, la distinction, qui rendrait compte de la différence entre une cession pure et simple et une mise à disposition des droits, est celle entre cession exclusive et non exclusive. Dès lors, la licence libre/open source, peut être considérée comme un contrat innomé qui s'analyse, comme le décrit Benjamin Jean<sup>92</sup>, en une cession non exclusive de droits d'auteur.

Ainsi, les licences libres/open source, en tant que contrat innomé, ne sont soumises qu'aux dispositions d'ordre public du droit des contrats et du droit d'auteur. Mais cette situation a suffi à soulever des interrogations quant à leur validité.

**33 – Les interrogations posées.** Ces licences sont issues de la pratique américaine, il était donc assez prévisible que leur appréhension par le droit français suscite des difficultés. Ces difficultés ont les retrouvent aussi bien dans les licences libres que dans les licences open source, dans la mesure où, s'il existe de multiples licences, toutes contiennent le minimum de droits prévus par les définitions respectives des deux mouvements et un minimum d'obligation, à savoir respect du droit de paternité de l'œuvre et exclusion de garantie et de responsabilité.

<sup>88</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°225, p. 147.

<sup>89</sup> CLEMENT-FONTAINE Mélanie, *Les œuvres libres*, *op. cit.*, n° 58, p. 50.

<sup>90</sup> LUCAS André et Henri-Jacques, *op. cit.*, n°567, p. 430.

<sup>91</sup> *Idem.*

<sup>92</sup> JEAN Benjamin, « *Option libre: compatibilité entre contrats* », mémoire de Master 2 sous la direction de Monsieur le Professeur Michel VIVANT, Montpellier I / CNRS, 2006, p.10.

Sur la forme du contrat, on peut citer le respect du formalisme imposé par le Code de la propriété intellectuelle pour les contrats d'auteur. L'article L 131- 3 exige que « *le domaine d'exploitation des droits cédés soit délimité quant à son étendue et à sa destination, quant au lieu et quant à la durée* ». Or, qu'il s'agisse de licence libre ou de licence open source, la plupart n'atteignent pas ce niveau de précision<sup>93</sup>. Toutefois, la sanction encourue étant seulement une nullité relative<sup>94</sup>, puisque le fondement de la règle est la protection de l'auteur, il semble que cette difficulté ne soit pas trop épineuse. Néanmoins, il existe un risque juridique, puisque rien ne garanti que l'auteur ne souhaite pas revenir sur son autorisation.

S'agissant des clauses, on peut citer les clauses d'exclusion de garantie et de responsabilité. Les clauses d'exclusion de responsabilité sont admises en droit français, elles ne vont donc pas invalider les licences. Cependant, elles ne recevront pas plein effet dans certaines circonstances<sup>95</sup> : l'exclusion ne vaut pas en cas de faute lourde ou de dol, puisque cette règle est d'ordre public. De même, pour les clauses d'exclusion de garantie, si la garantie des vices cachés n'est pas impérative et peut être exclue compte tenu de la nature du contrat<sup>96</sup>, il en va autrement de la garantie d'éviction. La garantie du fait des tiers<sup>97</sup> et du fait personnel semble, en effet, avoir valeur de principe général<sup>98</sup> en matière de droits d'auteur.

<sup>93</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit*, n°199, p.127.

<sup>94</sup> Voir Paris, 20 déc. 1989, D 1990, IR 32.

<sup>95</sup> Le régime des clauses abusives a également été invoqué mais la doctrine estime que, comme une clause abusive est une clause qui déséquilibre de manière significative les rapports entre les parties, une clause sera difficilement reconnue comme telle dans les licences libres/open source. Cela s'explique par l'économie du contrat : le licencié n'a que peu d'obligations alors que le donneur de licence met à disposition des droits étendus. Voir SEDALLIAN Valérie, « Garantie et responsabilité dans les logiciels libres », *Lamy droit de l'informatique et des réseaux*, n° 152, novembre 2002, p.8-16, <http://www.juriscom.net/pro/2/da20020901.pdf>.

<sup>96</sup> Les logiciels libres/open source ont justement vocation à faciliter la détection des failles du logiciel par l'ouverture du code source. La garantie des vices cachés est donc à l'opposé du concept de logiciel libre/open source.

<sup>97</sup> Bien que celle-ci pourrait être exclue du fait du caractère gracieux de la licence. HUET Jérôme, « La mise à disposition gratuite d'œuvres sur les réseaux numériques », in *Droit et technique*, LexisNexis, Litec, 2007, p.251.

<sup>98</sup> CROZE H. et BISMUTH Y., « Le contrat dit de licence de logiciel », JCP E 1986, II, n° 14659, n°426, p. 126.

Certaines licences libres/open source prévoit cette garantie<sup>99</sup>, mais la plupart restent muettes sur ce point. Il faut en fait considérer, qu'en cas de problème, la clause d'exclusion ne jouera pas et que le donneur de licence en sera tenu<sup>100</sup>.

Enfin, le dernier exemple que l'on peut citer est celui de l'autorisation de modifier le logiciel. La question de la modification d'une œuvre relève du droit à l'intégrité de l'œuvre, droit moral par principe inaliénable. Or, « *l'inaliénabilité du droit au respect de l'œuvre, principe d'ordre public, s'oppose à ce que l'auteur abandonne au cessionnaire, de façon préalable et générale, l'appréciation exclusive des utilisation, diffusion, adaptation, retrait, adjonction et changement auxquels il plairait à ce dernier de procéder* »<sup>101</sup>. Néanmoins, rappelons que la modification est envisagée parmi les droits patrimoniaux, en matière de logiciel, et que le droit à l'intégrité de l'œuvre se limite à interdire les modifications portant atteintes à l'honneur et à la réputation de l'auteur. Cet obstacle est donc moindre et ne remet pas en cause la validité des licences, tout au plus rappelle-t-il que l'auteur n'est pas dessaisi de ses droits<sup>102</sup>.

Finalement, beaucoup d'interrogations se sont posées mais, comme le remarque Yorick Cool, elles « *ont trouvé des réponses, et celles-ci sont rassurantes* »<sup>103</sup>.

D'ailleurs, les tribunaux français, comme les tribunaux étrangers, ont reconnu l'applicabilité de ces licences.

---

<sup>99</sup> C'est le cas de l'OSL, qui dans son article 7 énonce : "Licensor warrants that the copyright in and to the Original Work and the patent rights granted herein by Licensor are owned by the Licensor or are sublicensed to You under the terms of this License with the permission of the contributor(s) of those copyrights and patent rights". Voir <http://www.opensource.org/licenses/osl-3.0.php>.

<sup>100</sup> SEDALLIAN Valérie, *op. cit.*, p. 15.

<sup>101</sup> Cour de Cassation, Civ. 1<sup>ère</sup>, 28 janvier 2003, *Propr. Int.* 2003, 7, p.165, obs. P. Sirinelli.

<sup>102</sup> De plus, comme le remarque Fabrice de PATOUL, il existe une obligation d'exécuter les conventions de bonne foi qui devrait empêcher le donneur de licence d'abuser de ses prérogatives morales.

Voir COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°173, p. 108.

<sup>103</sup> *Idem*, n°304, p.190.

## 2 – La réception par la jurisprudence.

34 – **Reconnaissance des licences libres/open source.** Dans trois affaires<sup>104</sup> les tribunaux français<sup>105</sup> ont eu à connaître des licences libres/open source. Si dans aucune d’entre elles la question de la validité n’a été soulevée, on remarque que les juges ont considéré, à chaque fois, que le non-respect des dispositions de la licence entraînait la résolution du contrat conclu entre licencié et tiers acquéreur<sup>106</sup>. Cela implique donc une reconnaissance de ces licences<sup>107</sup>. On objectera que, dans les trois affaires, la licence en cause était une licence GPL, soit une licence mise en place par la Free Software Foundation. Néanmoins, cette reconnaissance pourrait très bien s’appliquer aux licences open source, puisque ces dernières, comme nous venons de le voir, reposent sur le même concept. De plus, la licence GPL est une licence certifiée « open source » par l’Open Source Initiative.

Les tribunaux ont également relevé un point important, celui de l’obligation d’information. En effet, il découle des décisions rendues que toute partie à un contrat fournissant un élément composé de logiciels libres/open source doit informer clairement son cocontractant de la présence d’un tel logiciel et du type de licence associée<sup>108</sup>. Ce qui était essentiellement visé par les juges, ce sont les effets de la clause de *copyleft*. En effet, ils existent de multiples licences, et l’un des points les plus importants à déterminer est celui de l’existence ou non d’une clause de *copyleft*, cela qu’il s’agisse d’une licence libre ou open source.

---

<sup>104</sup> CA Paris, 16 sept. 2009, « SA EDU4 c/ Association AFPA » ; TGI Chambéry, 15 nov. 2007, « Espaces et Réseaux Numériques c/ Conseil général de Savoie et Université de Savoie » ; TGI Paris, 28 mars 2007, « Educaffix c/ CNRS, Université Joseph Fourier et a. ».

<sup>105</sup> La France n’est pas la seule. Des tribunaux étrangers ont également eu à connaître ce type d’affaire, notamment un juge allemand qui a reconnu la violation des dispositions de la licence GPL par une société néerlandaise. Voir District court of Munich I, 19 nov. 2004, “Sitcom c/ Harald Welte”.

<sup>106</sup> On remarque que les sanctions sont celles de la responsabilité contractuelle et non celles de la contrefaçon, mais ceci s’explique par le fait que les auteurs des logiciels n’étaient pas partie aux litiges.

<sup>107</sup> RAMBAUD Sandrine, « Le juge français et le logiciel libre », *RLDI*, nov. 2009, n°54.

<sup>108</sup> *Idem*.

## Chapitre 2 – Des outils juridiques similaires

35 – Le fait d'utiliser le mécanisme contractuel permet de s'adapter aux besoins de chacun. S'il existe un socle commun à toutes les licences libres/open source, diverses clauses peuvent être rajoutées en fonction de la volonté du donneur de licence. De ce fait, chacun peut créer sa propre licence à condition de respecter les critères de la Free Software Foundation et/ou de l'Open Source Initiative. Toutefois, on remarque que, pour l'essentiel, les licences utilisées sont des contrats-types élaborés par diverses personnes morales et qui ont fait l'objet d'une approbation par la Free Software Foundation et/ ou l'Open Source Initiative. Cela est sensé simplifier l'identification d'un logiciel libre/open source et de son régime juridique. Mais même au sein de ces contrats-types, il existe une grande diversité (Section I).

Cette diversité pose des problèmes d'incompatibilité<sup>109</sup>, mais là n'est pas notre propos. Ce qu'il est intéressant de constater, c'est que cette diversité, se retrouve aussi bien dans le mouvement libre que dans le mouvement open source (Section II), par conséquent les deux mouvements proposent le même type d'outils juridiques.

### Section I – La diversité des licences

36 - Plusieurs classifications des licences ont été proposées (A), mais celle qui va retenir notre attention est la distinction entre licences permissives et licences copyleftées (B). C'est en effet sur cette distinction que l'on trouve, parfois, des confusions sur ce que sont les licences libres et les licences open source.

---

<sup>109</sup> Pour plus de précision sur cette question voir : JEAN Benjamin, « *Option libre : compatibilité entre contrats* », op. cit.

## A – Les différentes classifications

37 – Outre la classification licences permissives/ licences copyleftées, il existe une classification « historique »<sup>110</sup>, qui permet de comprendre la diversité par le contexte d'élaboration des licences (1), et une classification, proposée par Mélanie Clément-Fontaine<sup>111</sup>, en fonction du degré de liberté accordé (2).

### 1 – La classification « historique »

38 – **Les licences GNU.** L'une des premières licences libres, et la plus connue, est celle créée par Richard Stallman et la Free Software Foundation en 1989 : la *GNU General Public Licence*<sup>112</sup>.

Cette licence régit aujourd'hui la plus grande partie des logiciels libres, dont le système d'exploitation GNU/Linux pour lequel elle a été créée. Elle a été modifiée à deux reprises : la version 2 a vu le jour en 1991 et la version 3 en 2007.

Cette licence étant une licence copyleftée, le cas des bibliothèques logicielles posait des difficultés lorsque celles-ci étaient soumises à la GPL<sup>113</sup>. Pour résoudre ce problème, la Free Software Foundation a conçu une licence moins restrictive, la *GNU Lesser General Public Licence*<sup>114</sup> (LGPL) en 1991. Enfin, cette famille de licence est

<sup>110</sup> Classification proposée par Benjamin JEAN. Voir *Petit guide à l'usage des licences libres*, Matinée juridique du Syntec Informatique du 14 mars 2008, p.6.

Elle a également été envisagée par Mélanie Clément Fontaine dans sa thèse « Les œuvres libres » en terme de « critère identitaire », où elle estime qu'on peut classer les licences comme suit : celles rédigées par des personnes publiques, celles rédigées par des personnes privées du monde associatif et celles rédigées par des personnes privées du monde des affaires.

<sup>111</sup> CLEMENT-FONTAINE Mélanie, *Les œuvres libres*, op. cit.

<sup>112</sup> C'est en tout cas la première licence libre générique. Elle est en fait issue de la *GNU Emacs GPL*, dans laquelle Richard Stallman a retiré les mentions relatives au logiciel Emacs sur les conseils de John Gilmore. COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°6, p. 6.

<sup>113</sup> Le rôle d'une librairie est d'être liée à d'autres programmes dans le but de créer un logiciel exécutable. Si on interprète la GPL en considérant que l'effet contaminant du copyleft oblige à soumettre à la GPL tout logiciel résultant de l'utilisation d'une librairie copyleftée, cela pose problème aux développeurs.

Voir LEMYRE Pierre-Paul et WILLEMANT Richard, *Problématiques juridiques des logiciels libres et ouverts : défis et solutions pour le gouvernement du Québec (RESOLL)*, Rapport de projet, Montréal, Avril 2005, p. 24.

<sup>114</sup> Disponible sur : <http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html>.

complétée par la *GNU Free Documentation Licence*<sup>115</sup> (GFDL), conçue pour couvrir la documentation qui accompagne les logiciels mais qui peut être utilisée pour toute sorte de documents.

Cette famille de licences, conçue par la Free Software Foundation, est empreinte de la philosophie de cette dernière. Elles ont, notamment, toutes un *copyleft* plus ou moins fort.

39 – **Les licences académiques.** Parmi ces licences, on trouve la *Berkeley Software Distribution*<sup>116</sup> (BSD), rédigée par l'université de Berkeley et utilisée, entre autre, par les différentes variantes du système d'exploitation BSD Unix. On trouve également la licence MIT<sup>117</sup> rédigée par le Massachusetts Institute of Technology, pratiquement identique à la licence BSD.

Ces licences sont parmi les plus permissives, ce qui correspond à la vision des universités qui est de propager le savoir en restreignant le moins possible les libertés de l'utilisateur. Les obligations sont réduites au minimum, c'est-à-dire au respect du droit à la paternité et à l'exclusion de garantie et de responsabilité.

40 – **Les licences communautaires.** Ce sont des licences rédigées pour accompagner des projets de logiciels libres/open source particuliers. Elles répondent donc à des besoins spécifiques. C'est le cas de la licence Apache<sup>118</sup>, développée par l'Apache Software Foundation pour son serveur web.

41 – **Les licences institutionnelles.** Il s'agit de licences élaborées par des entreprises qui développent des logiciels libres/open source. De fait, ce sont les licences les plus précises et complètes. On y trouve la *Mozilla Public Licence*<sup>119</sup> (MPL),

---

<sup>115</sup> Disponible sur : <http://www.gnu.org/licenses/fdl.html>.

<sup>116</sup> Disponible sur : <http://www.opensource.org/licenses/bsd-license.php>.

<sup>117</sup> Disponible sur : <http://www.opensource.org/licenses/mit-license.php>.

<sup>118</sup> Disponible sur : <http://www.opensource.org/licenses/apache2.0.php>.

<sup>119</sup> Disponible sur : <http://www.opensource.org/licenses/mozilla1.1.php>.

qui constitue une sorte de compromis entre les licences BSD et GPL au regard de l'effet contaminant de la licence.

## 2 – La classification analytique

42 – **Licences offrant une liberté pérenne.** Ces licences règlent la question du sort de l'œuvre copiée, garantissant que son usage demeure libre. Ce sont des licences copyleftées.

43 – **Licences offrant une liberté fragile.** Ces licences ne s'intéressent pas au sort des modifications et à l'usage de l'œuvre. Elles se caractérisent le plus souvent par trois clauses. L'une rappelle les libertés octroyées, la seconde tend à ne pas permettre l'utilisation de l'œuvre à titre promotionnel, et la troisième fixe une clause de non-garantie<sup>120</sup>. C'est le cas, par exemple, de la licence BSD et de la licence Apache.

44 – **Licences offrant une liberté asymétrique.** Dans ces licences, l'auteur n'est pas contraint de la même manière que l'utilisateur. Les modifications apportées ne sont pas nécessairement sous la même licence que l'œuvre initiale. Mais l'auteur peut aussi se réserver des utilités qu'il refuse à l'utilisateur et combiner une licence libre et une exploitation commerciale, lesquelles peuvent, le cas échéant, reprendre des modifications apportées par des utilisateurs de la chaîne<sup>121</sup>. Ces licences sont généralement rédigées par des entreprises. C'est le cas de la *Netscape Public Licence*, qui exige que les modifications soient soumises à la même licence, mais qui réserve à la firme Netscape le droit de se soustraire à cette obligation pour ses propres développements<sup>122</sup>.

Ces classifications montrent que la diversité des licences libres/open source tient à des contextes d'élaboration divers et à des besoins différents en droits et

---

<sup>120</sup> Conseil supérieur de la propriété littéraire et artistique, *op. cit.*, p. 8.

<sup>121</sup> *Idem.*

<sup>122</sup> CLEMENT-FONTAINE Mélanie, *Les œuvres libres*, *op. cit.*, p. 75-104.



obligations que la licence doit contenir. Mais dans chacune des catégories, on trouve des licences qui répondent aussi bien aux critères de la Free Software Foundation qu'à ceux de l'Open Source Initiative<sup>123</sup>. De plus, aucune classification libre/open source n'a été proposée. On constate donc que les deux mouvements proposent les mêmes outils juridiques. Pour mieux s'en convaincre, il convient d'analyser la classification dite « classique » des licences, c'est-à-dire celle qui oppose licences permissives et licences copyleftées.

## B – Licences permissives et licences copyleftées

45 – Pour comprendre cette classification, il faut tout d'abord définir la notion de *copyleft* (1). Cette définition permettra ensuite de déterminer ce qu'est une licence permissive (2).

### 1 – La notion de *copyleft*

46 – **Garantie de la liberté.** Le terme *copyleft* a été choisi par opposition au *copyright*, équivalent anglophone du droit d'auteur. Il s'agit du mécanisme le plus original du mouvement libre, puisqu'il utilise la protection du droit d'auteur pour garantir la liberté. C'est pourquoi on a pu parler d'une utilisation du « *droit d'auteur contre le droit d'auteur* »<sup>124</sup>.

L'idée est que, sous couvert de la licence, on oblige l'utilisateur à distribuer le logiciel sous la même licence que celle dont il a bénéficié, ou, au moins, sous une licence qui offre autant de liberté. Ainsi, on évite, de façon définitive, toute exploitation propriétaire du code source, d'autant plus que ce mécanisme a un effet contaminant à l'égard des modifications opérées.

<sup>123</sup> Voir tableaux des licences, Annexe 3.

<sup>124</sup> LAURENT Philippe, Logiciels libres: "Le droit d'auteur contre le droit d'auteur", V.A.O. in intellectuelle rechten, 2003.

47 – **Statut juridique.** Selon Yorick Cool<sup>125</sup>, le *copyleft* peut être regardé, suivant les différentes rédactions, soit comme un pacte comissoire, soit comme une simple modalité d'exercice du droit de redistribution du licencié.

En effet, certaines des licences copyleftées stipulent que le contrat sera résolu en cas de non respect de la clause de copyleft<sup>126</sup>. Si on analyse de telles clauses en pactes comissoires, on peut alors considérer que le créancier a le choix entre demander la résolution du contrat et demander au débiteur d'exécuter son obligation. Cette dernière solution paraît plus conforme à l'esprit des licences, qui ont pour objectif la diffusion du code source des modifications.

Mais toutes les licences ne prévoient pas de telles conséquences<sup>127</sup>. Certaines se bornent à énoncer l'obligation du licencié de diffuser le logiciel et ses modifications sous la même licence, ce qui constitue une simple modalité de redistribution. Toutefois, cela montre les implications possibles pour le licencié, qui peut se voir privé des libertés qui lui ont été accordées.

Le licencié n'est pas le seul concerné, l'objectif étant que toute personne bénéficie des mêmes libertés sur le logiciel, les tiers doivent pouvoir invoquer une telle clause à leur profit. Pour Yorick Cool<sup>128</sup>, cela conduit à analyser le *copyleft* comme une stipulation pour autrui. La stipulation pour autrui se définit comme le « *contrat par lequel une personne, appelée stipulant, obtient d'une autre, le promettant, qu'elle exécute une prestation au profit d'une troisième appelée tiers bénéficiaire* »<sup>129</sup>. Une telle qualification permet de considérer que le tiers tire de la licence un droit direct et personnel envers le licencié, ce qui lui permet de lui réclamer l'exécution de son obligation sans passer par le donneur de licence.

<sup>125</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n°235, p.151.

<sup>126</sup> C'est le cas de la GPL v. 3 qui stipule dans son article 8 : « *You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License* ». Voir <http://www.gnu.org/licenses/gpl.html>.

<sup>127</sup> Par exemple, la *Q Public Licence*.

<sup>128</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n° 243 et s.

<sup>129</sup> GUILLIEN Raymond, VINCENT Jean, et al., *Lexique des termes juridiques*, 15<sup>e</sup> éd., Dalloz 2005, p. 587.

Cependant, si cette analyse nous paraît pouvoir s'appliquer, rappelons que les tribunaux français n'ont, à ce jour, prononcé que la résiliation du contrat passé entre le licencié et le tiers acquéreur en guise de sanction pour le non-respect de la clause de *copyleft*.

48 – **Déclenchement.** Le licencié n'a aucune obligation de modifier le logiciel et de publier ses modifications<sup>130</sup>. Or, le mécanisme du copyleft ne joue que s'il y a distribution. Du coup, la notion de distribution représente un enjeu important<sup>131</sup>.

En témoigne l'affaire « Free ». Des auteurs de logiciels libres accusent le fournisseur d'accès internet de distribuer des logiciels libres, soumis à la licence GPL, présents dans les décodeurs Freebox, en violation des obligations de la licence, puisque les codes sources ne sont pas rendus accessibles aux utilisateurs. La société Free, pour sa défense, estime qu'elle échappe à l'obligation de communication des codes sources dérivés dans la mesure où elle ne vend pas la Freebox aux abonnés, mais la loue, et qu'il n'y a, ainsi, pas eu distribution. En revanche, les auteurs des logiciels en question estiment que l'obligation s'applique quel que soit le mode de distribution<sup>132</sup>. Si, lors de l'écriture de ces lignes, la question n'a pas été tranchée par les tribunaux, la décision pourrait avoir un impact conséquent.

On peut également signaler des techniques de contournement, telle celle consistant pour le licencié à fournir un service via internet en utilisant le logiciel. Dans un tel cas de figure, il n'y a pas distribution du logiciel. Le destinataire du service n'obtient que le résultat du fonctionnement du logiciel. Ce type de pratique conduit les rédacteurs des licences à élargir la notion de distribution<sup>133</sup>. Par exemple,

<sup>130</sup> Hormis quelques exceptions. Voir infra n° 102.

<sup>131</sup> Sur les situations qui peuvent poser des difficultés voir également : TRIAILLE Jean-Paul, « Licences « open source » et contrats avec les développeurs et les distributeurs », *RDTI*, Lancier 2005, p. 65.

<sup>132</sup> TELLIER-LONIEWSKI Laurence et REVEL de LAMBERT Florence, « Logiciels libres: reconnaissance de la validité de la licence GNU GPL en France », note sous CA Paris 16 septembre 2009, *Gaz. Pal.* 2010, n° 36-37, p. 35-37.

<sup>133</sup> JEAN Benjamin, « *Option libre : compatibilité entre contrats* », op. cit., p. 20 - 23.

l'*Open Software Licence* préfère parler d'« *external deployment* », notion qui inclut les simples utilisations du logiciel au bénéfice d'une personne tierce<sup>134</sup>.

En tous cas, le fait que le *copyleft* ne se déclenche que lorsque le licencié décide de publier ses modifications montre que, contrairement à ce que pensait Christophe Caron<sup>135</sup>, le droit de divulgation ne met pas en péril ce système. Les contributeurs ne pourront pas invoquer une atteinte à leur droit de décider de la communication ou non de leur œuvre. Seules les modalités de la divulgation s'en trouvent affectées, encore que, rien n'empêche le licencié de divulguer ses modifications aux conditions qu'il souhaite s'il les distribue de manière indépendante de l'œuvre originale<sup>136</sup>. De plus, si le droit français définit le droit de divulgation comme le droit de décider du moment mais aussi des modalités de la communication de l'œuvre au public, on peut légitimement penser qu'une telle clause puisse être validée, puisqu'il s'agit d'une condition à l'autorisation donnée par l'auteur originaire pour utiliser son œuvre, juste contrepartie de la liberté de modification offerte<sup>137</sup>.

49 – **Effets.** L'effet contaminant du *copyleft* veut que la redistribution des œuvres dérivées soit effectuée sous les termes de la même licence que le logiciel original. Mais il existe une gradation dans les effets du *copyleft*. Ainsi, suivant la définition d'« œuvres dérivées » que retient la licence, on va trouver un *copyleft* standard ou un *copyleft* fort.

---

<sup>134</sup> Article 5 de l'OSL : « *The term "External Deployment" means the use, distribution, or communication of the Original Work or Derivative Works in any way such that the Original Work or Derivative Works may be used by anyone other than You, whether those works are distributed or communicated to those persons or made available as an application intended for use over a network. As an express condition for the grants of license hereunder, You must treat any External Deployment by You of the Original Work or a Derivative Work as a distribution under section 1(c).* ».

<sup>135</sup> CARON Christophe, « Les licences de logiciels dits « libre » à l'épreuve du droit d'auteur français », op. cit.

<sup>136</sup> Voir infra n° 49.

<sup>137</sup> Point sur lequel nous rejoignons l'analyse de Benjamin Jean. Voir « *Option libre : compatibilité entre contrats* », op. cit., p. 10. Mais également, COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., op. cit, n° 203, p. 131. Encore une fois, l'économie du contrat doit être prise en compte.

Les licences comprenant un *copyleft* fort sont celles qui considèrent comme œuvres dérivées les œuvres issues de la modification du code originaire et les œuvres qui incorporent ce code, modifié ou non<sup>138</sup>.

Seules les œuvres qui ne reprennent pas ce code et qui, même si elles sont distribuées au côté du programme soumis à la licence, peuvent être considérées comme des œuvres indépendantes et séparées, ne subissent pas les effets d'un tel *copyleft*.

Toutefois, la licence GPL, qui est sans doute la licence disposant de l'effet viral le plus extensif, considère que si ces œuvres sont distribuées dans un ensemble où elles interagissent avec le programme soumis à la licence, de telle façon que cet ensemble ne forme qu'un seul et même programme aux yeux de l'utilisateur final, elles doivent subir l'effet viral<sup>139</sup>. C'est le cas, par exemple, des bibliothèques, plug-ins et drivers<sup>140</sup>.

En revanche, les licences, qui ne contiennent qu'un *copyleft* standard, se limitent à soumettre aux mêmes termes les seules œuvres basées sur le programme original, c'est-à-dire celles qui sont issues de la modification du code soumis à la licence. C'est le cas, par exemple, de la *Mozilla Public Licence*, qui autorise l'intégration du code dans une œuvre plus large et qui ne soumet à la licence que le code en question et ses modifications, le reste de l'œuvre pouvant relever d'une licence différente<sup>141</sup>.

---

<sup>138</sup> JEAN Benjamin, « *Option libre : compatibilité entre contrats* », idem, n° 63, p. 31.

<sup>139</sup> Article 5 de la GPL: “You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it” et “A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate”.

<sup>140</sup> Sur ces questions voir : JEAN Benjamin, « *Option libre : compatibilité entre contrats* », *op. cit.*, p. 38-40 ; et COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n° 135 et s.

<sup>140</sup> Idem, n° 63, p. 31.

<sup>141</sup> LEMYRE Pierre-Paul et WILLEMANT Richard, *Problématiques juridiques des logiciels libres et ouverts : défis et solutions pour le gouvernement du Québec*, *op. cit.*, p. 24.

## 2 – Les licences permissives

50 – **Indifférence aux modifications.** A l'inverse, les licences permissives ne s'intéressent pas au sort des modifications. Tout licencié peut modifier le logiciel et le redistribuer sous la licence de son choix, y compris une licence propriétaire.

De même, il peut intégrer le programme dans une œuvre plus large soumise entièrement à une autre licence. Signalons également que ces licences permettent de redistribuer le logiciel non modifié sous des termes plus restrictifs, mais cette possibilité ne présente, à notre sens, que peu d'intérêt puisque le logiciel restera toujours disponible pour les tiers sous la licence d'origine, licence qui sera privilégiée puisque plus avantageuse<sup>142</sup>.

51 – **Des obligations limitées.** En fait, ces licences se contentent d'énumérer les libertés du licencié et n'impose que peu d'obligation<sup>143</sup>. Généralement seules les restrictions relatives au droit à la paternité et à l'exclusion de garantie y figurent. C'est d'ailleurs pour cela que l'on parle de ces licences comme des « licences de type domaine public », car l'étendue des droits rapproche le statut des œuvres soumises à ces licences à celui des œuvres tombées dans le domaine public. Toutefois, comme nous l'avons déjà signalé, ces œuvres restent protégées par le droit d'auteur.

La plus connue des licences permissives est la licence BSD, qui contient pour seules obligations celles de mentionner l'existence d'un droit d'auteur sur le code source, le code binaire et la documentation, de ne pas utiliser sans permission le nom du donneur de licence pour endosser ou promouvoir les versions modifiées du logiciel, et de reconnaître que le logiciel est fourni sans garantie.

On constate alors que la distinction n'est pas négligeable pour quiconque souhaite contribuer au logiciel. Les licences copyleftées et les licences permissives ne

---

<sup>142</sup> Point sur lequel nous rejoignons JEAN Benjamin. Voir : « Option libre : compatibilité entre contrats », op. cit., p. 19.

<sup>143</sup> JEAN Benjamin, *Petit guide à l'usage des licences libres*, op. cit., p. 6.

répondent pas aux mêmes besoins. Or, c'est pour cette raison que l'on a tendance à penser que les licences copyleftées relèvent uniquement de la Free Software Foundation, alors que les licences permissives relèveraient de l'Open Source Initiative. Cependant, nous allons voir que la situation est quelque peu différente.

## **Section II – Une diversité partagée par les deux mouvements**

52 – Si on considère souvent que le *copyleft* est un concept propre à la Free Software Foundation<sup>144</sup>, on constate en fait qu'il est également reconnu par l'Open Source Initiative (A). De même, si on a tendance à considérer que les licences permissives sont des licences open source, on remarque que ce sont aussi des licences libres (B). Remarquons qu'un tel constat est possible car nous avons choisi d'écarter toutes considérations idéologiques. Il est tout à fait juste de dire que les licences permissives correspondent davantage à la vision du mouvement open source et les licences copyleftées à celle du mouvement libre.

### **A – Le copyleft : un concept rattaché à la Free Software Foundation mais reconnu par l'Open Source Initiative**

53 – Nous allons, dans un premier temps, déterminer les raisons qui amènent à rattacher le mécanisme du *copyleft* aux licences libres (1), puis nous verrons que, toutefois, les licences copyleftées peuvent être open source (2).

---

<sup>144</sup> Voir par exemple : HUET Jérôme, « La mise à disposition gratuite d'œuvres sur les réseaux numériques », in *Droit et technique*, LexisNexis, Litec, 2007, p. 244.

## 1 – Un concept rattaché au mouvement libre

54 – **Un concept inventé par la Free Software Foundation.** Le terme *copyleft* est une idée de Don Hopkins<sup>145</sup> pour désigner le mécanisme juridique imaginé par Richard Stallman dans le cadre du projet GNU et généralisé avec la création de la Free Software Foundation et de la licence GPL. Historiquement, ce concept est donc issu de la pensée du fondateur de la Free Software Foundation et, par conséquent, il est lié à la philosophie de ce mouvement. En effet, comme nous le verrons<sup>146</sup>, l'objectif de la Free Software Foundation est avant tout d'assurer le partage de la connaissance. Or, le *copyleft* permet de garantir la pérennité d'un tel partage, puisqu'il interdit l'appropriation du code sous licence libre et de ses modifications<sup>147</sup>. Dans la vision de Richard Stallman, le développement des logiciels libres ne peut s'appuyer que sur des licences copyleftées. Ainsi, toutes les licences créées par la Free Software Foundation sont des licences copyleftées. Cela explique que l'on assimile les licences copyleftées aux licences libres.

55 – **Un concept problématique pour les entreprises.** Cette assimilation peut également s'expliquer par le fait qu'un tel mécanisme pose quelques difficultés aux entreprises, au-delà même du fait qu'il est présenté comme un instrument de lutte contre la propriété intellectuelle<sup>148</sup>. Le *copyleft* étant un concept général qui se manifeste de façon différente selon les licences, il entraîne parfois un sentiment d'insécurité.

---

<sup>145</sup> « En 1984 ou 1985, Don Hopkins (dont l'imagination était sans bornes) m'a envoyé une lettre. Il avait écrit sur l'enveloppe plusieurs phrases amusantes, et notamment celle-ci : « Copyleft - all rights reversed. ». J'ai utilisé le mot « copyleft » pour donner un nom au concept de distribution que je développais alors », Richard Stallman, « Le projet GNU », Site de l'organisation GNU, <http://www.gnu.org/gnu/thegnuproject.fr.html>.

<sup>146</sup> Infra n° 87.

<sup>147</sup> « Le but du projet GNU était de rendre les utilisateurs libres, pas de se contenter d'être populaire. Nous avons besoin de conditions de distribution qui empêcheraient de transformer du logiciel GNU en logiciel propriétaire. Nous avons utilisé la méthode du copyleft, ou « gauche d'auteur », « Le projet GNU », idem.

<sup>148</sup> Si l'objectif est, effectivement, d'éviter toute appropriation des programmes développés en libre/open source, le *copyleft* n'est, cependant, pas aux antipodes du droit d'auteur, et plus largement de la propriété intellectuelle. En fait, c'est parce que le droit d'auteur permet à l'auteur de décider des conditions d'exploitation de son œuvre qu'une telle obligation peut exister.



Par exemple, comme nous l'avons vu, la licence GPL dispose d'un *copyleft* fort qui, a priori, contaminerait tout logiciel qui interagit avec une librairie soumise à cette licence. Néanmoins, des divergences d'interprétations existent sur ce point. La Free Software Foundation considère que ce phénomène se produit, que le lien entre logiciel et librairie soit statique<sup>149</sup> ou dynamique<sup>150</sup>, mais, pour Lawrence Rosen<sup>151</sup>, si le lien est dynamique, le *copyleft* ne doit pas jouer<sup>152</sup>. De fait, il n'y a pas d'interprétation définitive, ce qui peut soulever des problèmes.

De même, le fait pour les salariés, les développeurs indépendants à qui la société fait appel, ou encore le distributeur, d'utiliser des programmes soumis à des licences copyleftées, sans que l'entreprise en soit informée, peut avoir des conséquences importantes pour celle-ci<sup>153</sup>. Elle peut alors être contrainte de distribuer ses propres produits sous licence libre, et voir son *business model* s'effondrer. C'est d'ailleurs pour cela que l'obligation d'information, rappelée par les tribunaux français<sup>154</sup>, est indispensable.

Ces difficultés peuvent conduire les entreprises à interdire le recours à des logiciels sous licence libre/open source et à inclure des clauses d'interdiction dans les contrats d'achat ou de développement.

Il faut également signaler que Eric Raymond, l'un des fondateurs de l'Open Source Initiative, estime que le *copyleft* est un mécanisme qui, finalement, est inutile, puisque les logiciels propriétaires étant théoriquement moins performants que les logiciels libres/open source, il n'est pas avantageux pour les utilisateurs d'approprier

---

<sup>149</sup> Le code de la librairie correspondant aux fonctions appelées est intégralement repris dans le logiciel.

<sup>150</sup> Seule une référence est inscrite dans le code exécutable du logiciel et cette référence permettra, le moment venu, d'appeler la fonction voulue dans la librairie. Dès lors, la librairie devra toujours figurer aux cotés du logiciel.

<sup>151</sup> Membre et conseil de l'OSI.

<sup>152</sup> Puisque le code n'est pas intégré au logiciel.

Pour plus de précision sur cette divergence voir : COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n° 139 et s.

<sup>153</sup> Cela peut la conduire à devoir distribuer un produit « propriétaire » sous licence libre/open source ou à devenir contrefactrice du programme libre/open source utilisé. Voir TRIAILLE Jean-Paul, « Licences « open source » et contrats avec les développeurs et les distributeurs », *op. cit.*, p. 54-68.

<sup>154</sup> Voir supra n° 34.

un logiciel. Pour lui, l'efficacité du système doit suffire à éviter ce type de comportement<sup>155</sup>.

Toutefois, cela ne signifie pas que les licences open source ne puissent pas être des licences copyleftées.

## 2 – Un concept présent dans le mouvement open source

56 – **Un concept conforme à l'*Open Source Definition***. Rappelons que pour déterminer si une licence est open source, il faut se référer à la définition donnée par l'Open Source Initiative, définition composée de dix critères. Si on analyse ces critères, on constate que toutes les licences copyleftées répondent à ces critères.

Toutefois, on a pu penser qu'une licence contenant un *copyleft* fort, telle que la GPL, ne répondait pas au critère numéro neuf, qui indique que la licence ne doit pas contaminer d'autres logiciels. Nous avons vu, en effet, que la licence GPL pouvait contaminer des logiciels distribués avec le programme qu'elle couvre, dans certaines circonstances. En l'occurrence, c'est ce qui semble interdit par l'*Open Source Definition* où il est énoncé que « *la licence ne doit pas apposer de restrictions sur d'autres logiciels distribués avec le programme qu'elle couvre* »<sup>156</sup>. Néanmoins, rappelons que cet effet du *copyleft* de la GPL ne vaut que si le programme peut être envisagé comme un tout et non lorsqu'il s'agit de distribuer des logiciels indépendants sur un même support<sup>157</sup>.

<sup>155</sup> Interview d'Eric Raymond par Frédéric Biancuzzi, « Nous n'avons plus besoin de la GPL », Site Framasoft, 30 juin 2005, <http://www.framasoft.net/article4128.html>.

<sup>156</sup> “The license must not place restrictions on other software that is distributed along with the licensed software”,

“The Open Source Definition”, Site de l'Open Source Initiative, <http://www.opensource.org/docs/osd>.

<sup>157</sup> Ce que la FSF appelle un agrégat. Voir article 5 de la GPL v.3 qui stipule : “A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.”.

Or, il nous semble que c'est bien cette dernière situation qui est en réalité visée par l'*Open Source Definition*<sup>158</sup>, puisque l'exemple choisi pour illustrer le critère est celui de programmes distribués grâce au même médium<sup>159</sup>. Dès lors, nous estimons que, quelque soit le type de *copyleft*, fort ou standard, un tel mécanisme ne fait pas obstacle à l'approbation d'une licence par l'Open Source Initiative.

D'ailleurs, si on observe la liste des licences open source donnée par l'Open Source Initiative, on remarque que toute la famille des licences GPL, élaborées par la Free Software Foundation, a été certifiée open source. Quant à l'*Open Software License*, rédigée par Lawrence Rosen et présentée par Yorick Cool et Philippe Laurent comme la licence de l'Open Source Initiative<sup>160</sup>, il s'agit également d'une licence disposant d'un *copyleft*, qui plus est un *copyleft* fort.

**57 – Une simple obligation.** Ceci peut s'expliquer par le fait que le *copyleft* est un outil juridique qui, en soit, ne représente rien d'autre qu'une obligation particulière pour le licencié. Objectivement, l'Open Source Initiative, dans son rôle d'approbation des licences, n'a aucune raison d'écarter les licences copyleftées, qui répondent tout aussi bien aux impératifs techniques de la méthode de développement open source, que les licences permissives. Au contraire, disposer d'outils juridiques variés favorise l'expansion des logiciels open source, puisque cela permet de répondre à des besoins différents, comme nous l'avons déjà remarqué. Les entreprises, elles-mêmes, peuvent trouver un intérêt dans l'utilisation d'un tel mécanisme<sup>161</sup>.

<sup>158</sup> En ce sens: JEAN Benjamin, *Petit guide à l'usage des licences libres*, op. cit., p. 4.

<sup>159</sup> "For example, the license must not insist that all other programs distributed on the same medium must be open-source software", "The Open Source Definition", Site de l'Open Source Initiative, <http://www.opensource.org/docs/osd>.

<sup>160</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., op. cit, n° 26, p. 17.

<sup>161</sup> Par exemple, les licences de Microsoft approuvées par l'OSI et la FSF sont des licences copyleftées. Nous pensons qu'une fois maîtrisé, le *copyleft* est un bon outil pour les entreprises qui souhaitent développer des logiciels libres/open source, puisqu'il permet d'éviter qu'un concurrent utilise le produit pour le commercialiser de façon propriétaire avec des améliorations. Ainsi, même s'il ne s'agit pas d'une logique d'appropriation, cette obligation n'est pas forcément néfaste pour l'entreprise. De plus, rappelons que le *copyleft* ne joue qu'en cas de distribution du logiciel, or beaucoup d'entreprises n'utilisent les logiciels libres qu'elles modifient qu'en interne.

Néanmoins, la vision pragmatique de l'Open Source Initiative peut conduire à des divergences d'interprétation sur la portée des clauses de *copyleft*. C'est le cas, comme nous l'avons vu, pour celle de la licence GPL. En effet, pour Lawrence Rosen, d'un point de vue purement technique, il n'y a aucune raison de considérer que le *copyleft* de cette licence doit jouer lorsqu'il existe seulement un lien dynamique entre librairie et logiciel, puisque la relation n'est que transitoire et que le programme qui fait appel à la librairie n'a pas à être modifié. En revanche, la Free Software Foundation, qui considère le *copyleft* comme une « arme » idéologique en faveur de la liberté, estime que le lien suffit à faire jouer le mécanisme<sup>162</sup>. Pour autant, ce ne sont que des indications quant à l'interprétation à retenir, et celle qui sera prise en compte par le juge, est celle du donneur de licence puisque le contrat doit être une transcription de sa volonté. Les licences étant généralement des contrats types proposés par des organismes, il lui est toujours possible d'éclaircir certains points en recourant aux interprétations. Ainsi, par exemple, Linus Torvalds a précisé que, concernant la licence GPL utilisée pour le noyau Linux, les programmes qui ne faisaient que graviter autour du noyau ne devaient pas être soumis à cette licence<sup>163</sup>.

Si on constate que les licences copyleftées peuvent être des licences libres comme des licences open source, il en va de même pour les licences permissives.

## **B – Les licences permissives : rattachées à l'Open Source Initiative mais reconnues par la Free Software Foundation**

58 – Si on peut présenter les licences permissives comme le modèle type des licences open source (1), ces licences n'en sont pas moins également des licences libres (2).

<sup>162</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n° 139 et s.

<sup>163</sup> JEAN Benjamin et ZAPOLSKY Alexandre, « La propriété intellectuelle dans l'industrie de l'open source », *Gaz. Pal.*, oct. 2008, n°298-299, p. 21.

## 1 – Des licences rattachées à l'Open Source Initiative

59 – **Indifférence à l'appropriation.** Le mouvement open source est présenté comme un mouvement pragmatique pour lequel le développement de logiciel libre/open source n'est pas un moyen de lutte contre le modèle propriétaire, mais simplement une alternative plus performante. De ce fait, contrairement à la Free Software Foundation, l'Open Source Initiative ne considère pas les logiciels propriétaires comme un problème, mais plutôt comme une solution sous optimale<sup>164</sup>. Dans une telle approche, le mélange entre logiciels libres et logiciels propriétaires est donc tout à fait envisageable dès lors que cela n'altère pas l'efficacité du programme. Or, c'est bien cette possibilité qu'offre les licences permissives. De plus, l'Open Source Initiative ayant été fondée pour intégrer le modèle open source dans le monde des affaires, on peut estimer que ce type de licence correspond mieux à cet objectif.

Toutefois, rappelons que les licences les plus permissives sont celles que l'on qualifie de « type domaine public ». Ces licences sont développées dans des contextes académiques dans l'optique de partager les logiciels développés avec le moins de conditions possibles, ce qui ne semble pas être à même de séduire les entreprises<sup>165</sup>. Au contraire, les sociétés auront plutôt tendance à préférer des licences plus complètes, plus exhaustives qui leur permettent de garder un certain contrôle sur leurs produits.

En fait, les licences de type domaine public semble se rapprocher de la vision de la Free Software Foundation, même s'il leur manque l'opportunité de garantir la liberté accordée.

---

<sup>164</sup> Voir infra n° 90 et s.

<sup>165</sup> Ainsi, nous ne partageons pas la vision de Mélanie Clément-Fontaine qui présente les licences offrant une liberté fragile comme étant les licences open source et les licences de type domaine public, et qui considère les secondes comme un sous-ensemble des premières. CLEMENT-FONTAINE Mélanie, *Les œuvres libres*, op. cit., p. 75-104.

## 2 – Des licences reconnues par la Free Software Foundation

60 – **Incitation à l’usage du *copyleft***. Le mouvement libre s’oppose à l’appropriation des logiciels, et en particulier des logiciels libres car il ne souhaite pas que l’industrie propriétaire profite de ses avancées<sup>166</sup>. La préoccupation principale du mouvement a, d’ailleurs, été de pouvoir rendre libre un logiciel mais aussi de savoir comment préserver cette liberté. C’est pourquoi toutes les licences créées par la Free Software Foundation sont des licences copyleftées. C’est aussi ce qui laisse penser que les licences permissives ne sont pas considérées comme libres par la Free Software Foundation.

Il est vrai que, lorsque l’on regarde les commentaires effectués par la Free Software Foundation sur les licences, on constate que celle-ci conseil régulièrement aux développeurs de ne pas utiliser des licences non-copyleftées, et de manière générale des licences autres que celles de la famille GNU, pour couvrir leurs logiciels<sup>167</sup>.

Toutefois, il s’agit là d’un idéal et cela ne signifie pas que les licences permissives ne sont pas des licences libres.

61 – **Une définition exempte d’obligations**. La Free Software Foundation, tout comme l’Open Source Initiative, publie une liste des licences disponibles qu’elle approuve ou non au regard de sa définition<sup>168</sup>. Or, contrairement à ce que l’on peut lire parfois, la définition d’un logiciel libre selon la Free Software Foundation ne retient que les libertés que l’utilisateur doit avoir vis-à-vis du logiciel. Il n’y a pas de conditions ou d’obligations.

<sup>166</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n° 4, p. 5.

<sup>167</sup> « Au sein du projet GNU, nous recommandons généralement l’utilisation de licences de type « *copyleft* » comme la GNU GPL plutôt que des licences de logiciels libres plus permissives », Richard Stallman, « Pourquoi le copyleft », Site de l’organisation GNU, <http://www.gnu.org/philosophy/why-copyleft.fr.html>

<sup>168</sup> Liste disponible sur : <http://www.gnu.org/licenses/license-list.fr.html>.

Ce que dit la Free Software Foundation sur le *copyleft*, c'est que « *certain types de règles sur la manière de distribuer le logiciel libre sont acceptables tant que ces règles ne rentrent pas en conflit avec les libertés fondamentales. Par exemple, le copyleft (pour résumer très simplement) est une règle qui établit que lorsque vous redistribuez les programmes, vous ne pouvez ajouter de restrictions pour retirer les libertés fondamentales au public. Cette règle ne rentre pas en conflit avec les libertés fondamentales ; en fait, elle les protège* »<sup>169</sup>. On remarque qu'en aucun cas elle n'impose une telle clause<sup>170</sup>.

Dès lors, si une licence offre les quatre libertés définies par la Free Software Foundation sur un logiciel, il s'agira d'une licence libre, peu importe qu'elle soit copyleftée ou non. Ainsi, la licence BSD, une des premières licences permissives, est considérée comme libre par la Free Software Foundation<sup>171</sup>. Cela est clairement exprimé dans les explications qui suivent la définition du logiciel libre : « *Dans le projet GNU, nous utilisons le « copyleft » pour protéger ces libertés. Mais des logiciels libres non-copyleftés existent aussi. Nous croyons qu'il y a de bonnes raisons qui font qu'il est mieux d'utiliser le copyleft, mais si votre programme est libre non-copylefté, nous pouvons tout de même l'utiliser* »<sup>172</sup>.

En fait, les licences permissives sont considérées comme éthiques par le mouvement libre, simplement elles ne vont pas au bout de la logique.

Cette situation peut également s'expliquer par le fait que tous les acteurs du libre ne sont pas farouchement opposés à toute appropriation<sup>173</sup>, ou encore parce que certains estiment que le *copyleft* crée un cloisonnement qui prive ceux qui ne sont pas en mesure de tenir les conditions des licences copyleftées, de pouvoir profiter des

<sup>169</sup> « Définition d'un logiciel libre », Site de l'organisation GNU, <http://www.gnu.org/philosophy/free-sw.fr.html>.

<sup>170</sup> En ce sens : REVILLARD Diane, DIERMARK, *Livre blanc organisations et logiciels libres*, op. cit., p. 42.

<sup>171</sup> « Les licences commentées », Site de l'organisation GNU, <http://www.gnu.org/licenses/license-list.fr.html>.

<sup>172</sup> « Définition d'un logiciel libre », idem.

<sup>173</sup> C'est le cas des universités, pour lesquelles le partage doit se faire sans conditions. Si la FSF avait imposé le copyleft, elle aurait exclu les licences créées par ce type d'acteur et se serait privé de leurs contributions au mouvement.

codes disponibles sous ces licences<sup>174</sup>. De ce fait, la Free Software Foundation, dans son rôle d’approbation des licences, doit faire preuve de flexibilité, ce qui passe par une tolérance pour ces licences permissives<sup>175</sup>. Il nous semble, alors, qu’il ne faut pas cantonner le concept de libre au mécanisme du *copyleft*, ce dernier n’étant qu’un des aspects du libre.

62 - *In fine*, les mouvements libres et open source ont pour objectif d’encourager le développement de logiciels que l’on peut librement utiliser, copier, distribuer et modifier. Pour cela, tous deux préconisent l’utilisation de licences. Ces licences sont diverses, tant du fait de leur origine que de leur contenu, mais cette diversité ne constitue pas un point de rupture entre libre et open source : la majorité des licences existantes peuvent être qualifiées de libre et d’open source. Cela s’explique par le fait que l’Open Source Initiative et la Free Software Foundation requiert un minimum qui correspond à leurs définitions respective et qui est quasiment identique. Au-delà, toutes les variations dans les droits et obligations qui sont conférés à l’utilisateur n’affectent pas, ou peu, l’approbation de la licence, tant que ces variations n’entravent pas ce minimum requis.

A ce stade, il nous est donc possible de dire, qu’a priori, on peut confondre libre et open source, qui ne sont que des synonymes pour un même concept.

Cependant, deux organisations existent et proposent des définitions différentes, ce qui laisse penser que, malgré tout, il existe bien une différence entre libre et open source. C’est pourquoi nous nous proposons, maintenant, d’étudier cette distinction.

---

<sup>174</sup> Alors que, pourtant, le principe du libre est de permettre l’accès à tous à ces codes. Voir LEMYRE Pierre-Paul et WILLEMANT Richard, *op. cit.*, p. 24.

<sup>175</sup> Toutefois, beaucoup considèrent que la FSF n’est toujours pas suffisamment flexible et utilise son autorité pour favoriser l’utilisation des licences GNU. En ce sens : COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n° 3, p. 3.



## Partie 2 – Logiciel libre et open source : une analogie imparfaite

63 - Richard Stallman a écrit un essai où il explique pourquoi il considère que parler de logiciel libre est plus juste que de parler de logiciel open source<sup>176</sup>.

Au regard de cet article, et des différentes prises de position des deux mouvements, on remarque que l'opposition passe d'abord par une divergence terminologique liée à la confusion suscitée aussi bien par le terme *free* que par le terme *open source* (Chapitre 1). Mais cette divergence, nous le verrons, n'est pas sans lien avec l'opposition idéologique entre les deux mouvements, opposition qui entraîne quelques différences pratiques entre logiciel libre et logiciel open source (Chapitre 2). Dès lors, l'analogie que nous avons observée n'est pas absolue.

### Chapitre 1 – L'opposition terminologique

64 - Comme nous l'avons dit, cette opposition se fonde sur la confusion suscitée à la fois par le terme *free* et par le terme *open source*.

En effet, chacun des deux termes ne permet pas de saisir immédiatement ce qu'est un logiciel libre (tout comme le terme libre d'ailleurs) et a pu entraîner des amalgames préjudiciables au mouvement du libre.

Nous étudierons, tout d'abord, quel problème pose le terme *free* (Section I), problème qui a entraîné l'apparition de l'appellation *open source*, ce qui nous amènera à étudier le problème posé par celle-ci (Section II).

---

<sup>176</sup> R. Stallman, « Pourquoi « logiciel libre » est-il meilleur que « open source » », Site de l'organisation GNU, <http://www.gnu.org/philosophy/open-source-misses-the-point.fr.html>.

## Section I – Le problème du terme *free*

65 - Le logiciel libre est un concept qui est né aux Etats-Unis et qui en anglais se dit *free software*, or le terme *free*, dans cette langue, signifie aussi bien libre que gratuit. Ainsi, beaucoup ont cru que ce nouveau concept faisait référence à des logiciels gratuits, d'autant plus qu'effectivement la majorité d'entre eux le sont.

Pour autant, la gratuité n'est pas une caractéristique du logiciel libre (A), et cet amalgame, qui a conduit à un rejet de ce concept par les entreprises, a montré la nécessité d'élaborer une nouvelle appellation : c'est l'apparition du terme *open source* (B).

### A – L'amalgame de la gratuité

66 – Si on a pu considérer les logiciels libres comme des logiciels gratuits du fait de leur dénomination en anglais, il semble que seule la licence soit toujours gratuite (1) alors que le logiciel peut être distribué à titre onéreux (2).

#### 1 – Une licence toujours gratuite

67 – **La gratuité admise par la loi.** La loi DADVSI du 1<sup>er</sup> août 2006, relative au droit d'auteur et aux droits voisins dans la société de l'information, a introduit dans le Code de la propriété intellectuelle un article L 122-7-1<sup>177</sup> qui autorise l'auteur à mettre gratuitement ses œuvres à la disposition du public. On peut estimer qu'il s'agit là d'une volonté du législateur de reconnaître l'existence des mises à disposition gratuites telles que le logiciel libre<sup>178</sup>.

---

<sup>177</sup> « L'auteur est libre de mettre ses œuvres gratuitement à la disposition du public, sous réserve des droits des éventuels coauteurs et de ceux des tiers ainsi que dans le respect des conventions qu'il a conclues. ».

<sup>178</sup> Il faut remarquer que cette disposition ne se trouve pas dans la directive CE n° 2001/29 du 22 mai 2001 sur l'harmonisation de certains aspects du droit d'auteur et des droits voisins dans la société de l'information, dont la loi DADVSI est pourtant la transposition.

Cependant, il faut remarquer que cette possibilité était déjà offerte par la législation, dans la mesure où elle correspond parfaitement à la logique de la propriété littéraire et artistique, qui est d'assurer à l'auteur la maîtrise de son œuvre.

En effet, l'article L 122-7 du Code de la Propriété Intellectuelle énonçait déjà que « *le droit de représentation et le droit de reproduction sont cessibles à titre gratuit ou à titre onéreux* ». Or, comme le précise Jérôme Huet, « *comme chacun le sait, qui peut le plus peut le moins et, si l'auteur, ce qui va sans dire d'ailleurs, est en droit de céder gratuitement son œuvre, il s'ensuit qu'il peut en faire profiter gracieusement autrui en la laissant librement à sa disposition* »<sup>179</sup>.

De plus, si le Code de la Propriété Intellectuelle contient des dispositions d'ordre public concernant la rémunération des auteurs dans les contrats portant sur les droits d'auteur, comme le fait remarquer Huguet « *les restrictions impératives apportées par la loi à la liberté de rémunération ne portent pas sur le principe même de la rétribution de l'auteur. [...] L'ordre public se manifeste dans le mode de rémunération de l'auteur* »<sup>180</sup>.

**68 – La gratuité utilisée dans les licences libres.** Il nous semble que les auteurs de logiciel libre utilisent précisément cette possibilité offerte par la législation. Effectivement, la licence, qui permet de mettre à disposition les droits de l'auteur, sera, selon nous, toujours gratuite. Ce qui signifie que la liberté d'exécuter, copier, distribuer, modifier le logiciel est donnée gratuitement<sup>181</sup>. Si l'on rémunérait l'auteur pour l'exercice de l'un de ces droits, alors on ne pourrait plus parler de liberté d'exercice de ces droits, sachant que la rémunération doit alors être fixée en fonction de l'exploitation qui est faite de l'œuvre, qu'elle soit proportionnelle ou forfaitaire<sup>182</sup>.

<sup>179</sup> HUET Jérôme, op. cit., 2007, p.240.

<sup>180</sup> HUGUET, « L'ordre public et les contrats d'exploitation du droit d'auteur », LGDJ, 1962, n°193.

<sup>181</sup> En ce sens : TRIAILLE Jean-Paul, op. cit., p. 54-68 ; LEMYRE Pierre-Paul et WILLEMANT Richard, op. cit., p. 16 ; et JEAN Benjamin, *Petit guide à l'usage des licences libres*, op. cit., p. 2.

<sup>182</sup> Article L 131- 4 du CPI.

C'est un principe que l'on retrouve dans les dix critères de l'*Open Source Definition* à l'article 1 sur la libre redistribution qui énonce que « *la licence ne doit pas exiger que cette vente (du logiciel) soit soumise à l'acquittement de droits d'auteur ou de royalties* »<sup>183</sup>.

Un autre point semble aller dans ce sens. Christophe Caron a écrit que « *le système contractuel imaginé, loin d'être une chaîne de contrats entre le donneur de licence, le licencié et ensuite de multiples sous-licenciés, est construit « en étoile »* »<sup>184</sup>, c'est-à-dire que l'on va trouver un donneur de licence au « centre » et, tout autour, de multiples licenciés. Si bien que c'est toujours la même personne qui accorde les droits sur l'œuvre originale.

Or, dans une licence libre, on permet aux licenciés de redistribuer le logiciel gratuitement mais aussi contre rémunération, même si le donneur de licence l'avait, à l'origine, distribué gratuitement. Ainsi, on peut avoir une distribution gratuite par le donneur de licence, puis une autre distribution gratuite par un licencié, puis une distribution payante par un autre licencié sans que la licence n'ait été modifiée. De plus, c'est le distributeur qui recevra le prix et non le donneur de licence<sup>185</sup>. Ce schéma ne semble s'appliquer qu'aux licences qui interdisent le sous-licencement<sup>186</sup>, mais cela n'empêche pas de considérer que toutes les licences libres sont des contrats à titre gracieux puisque, comme nous l'avons dit, le prix demandé pour un logiciel libre ne doit pas correspondre à l'acquittement de droits d'auteur.

Tout ceci tend donc à démontrer que la licence libre (et open source) sera toujours gratuite.

---

<sup>183</sup> “*The license shall not require a royalty or other fee for such sale*”, “The Open Source Definition”, Site de l’Open Source Initiative, <http://www.opensource.org/docs/osd>.

<sup>184</sup> CARON Christophe, *op. cit.*

<sup>185</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al., *op. cit.*, n° 223, p.146.

<sup>186</sup> C’est la cas de la plupart des licences copyleftées, telles que la GPL. Pour les licences qui ne l’interdisent pas, on va avoir une chaîne de contrats qui liera le tiers au licencié et non à l’auteur originaire. C’est le cas de la majorité des licences permissives telles que la licence BSD ou la licence MIT. JEAN Benjamin, « *Option libre : compatibilité entre contrats* », *op. cit.*, p. 11.

## 2 – La distribution à titre onéreux du logiciel

69 – « **Free as in free speech, not as in free beer** »<sup>187</sup>. Cette célèbre phrase de Richard Stallman a permis à celui-ci d'expliquer que le *free software* devait s'entendre d'un logiciel libre et pas obligatoirement d'un logiciel gratuit. Si la licence est toujours gratuite, en revanche la transmission du logiciel pourra être payante<sup>188</sup>. Il ne s'agit pas ici de payer pour exercer les droits sur l'œuvre, mais pour recevoir un exemplaire du logiciel.

Il y aurait ainsi deux opérations: la remise du logiciel et la mise à disposition des droits d'auteur par la licence<sup>189</sup>. On retrouve cela dans la GPL qui stipule que « *vous pouvez exiger paiement pour l'acte physique de remise d'une copie (...)* »<sup>190</sup>.

Lorsque l'on est en présence d'un logiciel incorporé dans un support physique (CD-Rom, disquette...), la distinction est assez simple puisque l'on peut recourir au principe d'indépendance de l'œuvre et de son support.

En effet, le Code de la Propriété Intellectuelle prévoit que le propriétaire du support n'est pas obligatoirement titulaire des droits de propriété intellectuelle sur l'œuvre, si bien que le transfert de la propriété matérielle n'induit pas celui de la propriété intellectuelle<sup>191</sup>. Ainsi, l'on doit distinguer la propriété incorporelle sur l'œuvre et la propriété corporelle sur le support de l'œuvre. Cette dissociation permet de mieux comprendre comment l'opération portant sur la fourniture de l'exemplaire du logiciel peut donner lieu au paiement d'un prix, alors que l'opération sur les droits d'auteur se fait à titre gracieux.

<sup>187</sup> Richard Stallman, Site de l'organisation GNU, <http://www.gnu.org/philosophy/free-sw.fr.html>.

<sup>188</sup> Il est également possible de prévoir le paiement de services consentis en marge de la mise à disposition des droits, comme par exemple la fourniture d'une garantie. Voir TRIAILLE Jean-Paul, op. cit., p. 13.

<sup>189</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 222, p.145.

<sup>190</sup> Article 3 al. 2: "*You may charge a fee for the physical act of transferring a copy (...)*". Plus généralement, la FSF, lorsqu'elle évoque la vente de logiciel libre, parle de vente d'une copie du logiciel. Voir « Vente des logiciels libres », Site de l'organisation GNU, <http://www.gnu.org/philosophy/selling.fr.html>.

<sup>191</sup> Article L 111-3 du CPI.

Mais les choses se complexifient lorsque le logiciel est dématérialisé et que sa transmission s'effectue par le téléchargement d'un fichier. Le prix ne peut alors plus correspondre au coût du support.

Cependant, il est toujours possible de dissocier l'acte de communication du logiciel et la mise à disposition des droits sur l'œuvre, puisque ce qui fait l'économie de l'opération, qui va pouvoir donner lieu à paiement, c'est la mise à disposition du logiciel en tant qu'outil<sup>192</sup>, opération qui pourrait avoir lieu même s'il s'agissait d'un logiciel non protégé par le droit d'auteur faute d'originalité<sup>193</sup>.

Comme le logiciel est alors détaché de tout support<sup>194</sup>, la qualification de vente semble ne plus convenir, le programme ne pouvant être considéré comme une chose sur laquelle il y aurait transfert de propriété. Pour H. Croze et Y. Bismuth, le logiciel « est un procédé permettant à celui qui le connaît d'obtenir d'un ordinateur un service » et ainsi il est plus « proche de la catégorie juridique des services », mais « d'un service capitalisé » puisque le programme est réalisé à l'avance<sup>195</sup>.

Ainsi, l'opération qui consiste à transmettre un logiciel sans support pourrait être qualifiée de contrat d'entreprise<sup>196</sup>, contrat indépendant du contrat de licence libre.

Rappelons, toutefois, que la fixation d'un prix pour la communication du logiciel n'est qu'une faculté offerte au donneur de licence comme aux licenciés.

Il est donc possible de trouver des logiciels libres payants, mais il est aussi possible de trouver des logiciels non libres gratuits. De ce fait, la gratuité ne peut

<sup>192</sup> « Les biens informatiques, comme logiciels ou systèmes-experts, présentent, en effet, ce trait singulier d'être, en tout pays, appréhendés, au titre du droit d'auteur ou du copyright, comme des œuvres de forme, dans le temps même qu'ils sont des œuvres directement fonctionnelles, « opératoires » ». VIVANT Michel, « L'informatique dans la théorie générale du contrat », Dalloz, 1994, p. 117.

H. Croze et Y. Bismuth, eux, différencient les contrats portant sur la création intellectuelle et ceux portant sur le résultat de cette création. Ici, il s'agirait du résultat. H. CROZE et Y. BISMUTH, « Le contrat dit de licence de logiciel », *op. cit.*, p.122.

<sup>193</sup> LUCAS André, DEVEZE Jean, FRAYSSINET Jean, *op. cit.*, n° 743, p.490 et COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 222, p.145.

<sup>194</sup> Signalons que même en présence d'un support, on peut considérer que la qualification de vente ne convient pas si l'on estime que le prix demandé correspond à la valeur de l'outil transmis et non pas à celle du support. La prestation caractéristique n'est alors plus le transfert de propriété sur le bien matériel.

<sup>195</sup> H. CROZE et Y. BISMUTH, *op. cit.*, p. 123.

<sup>196</sup> En ce sens : COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 224, p.146. On peut aussi le voir comme un contrat innomé : LUCAS André, DEVEZE Jean, FRAYSSINET Jean, *op. cit.*, p. 495.

servir de repère pour déterminer si un logiciel est libre ou non, et c'est bien cette difficulté que pose le terme *free*, d'autant plus que cet amalgame rend le concept de logiciel libre difficile à adopter pour les entreprises. C'est pour clarifier les choses que la Free Software Foundation a publié une définition précise du logiciel libre, mais Richard Stallman le reconnaît lui-même : ce n'est qu'une solution partielle<sup>197</sup>. C'est sur ce constat que l'appellation *open source* est apparue.

## B – L'apparition du terme *open source*

70 - Nous verrons que le code source a une certaine importance en matière de logiciels libres (1), ce qui justifie le choix du terme *open source* par l'Open Source Initiative (2).

### 1 – L'importance du code source

71 – **Logiciels gratuits mais non libres.** Trois types de logiciel ne doivent pas être confondus avec le logiciel libre, bien que ces logiciels soient gratuits.

Tout d'abord, il y a le *freeware*, traduit par les québécois par « gratuiticiel ». S'il n'y a pas de définition communément admise, on peut, cependant, dire qu'il s'agit d'un logiciel dont l'auteur permet l'usage et la copie gratuitement. Néanmoins, le code source de ce logiciel n'est pas accessible et l'auteur se garde le droit de commercialiser des versions ultérieures du programme<sup>198</sup>. Il manque à ces logiciels la liberté de les modifier et de les redistribuer pour appartenir à la catégorie des logiciels libres.

Dans le même ordre d'idées, on trouve les *sharewares*, qui sont en fait des *freewares* dont l'usage gratuit est à durée limitée. Passé un certain laps de temps, l'utilisateur devra payer pour continuer à utiliser le programme, ce qui est contraire

---

<sup>197</sup> Richard Stallman, « Pourquoi « logiciel libre » est-il meilleur que « open source » », Site de l'organisation GNU, <http://www.gnu.org/philosophy/open-source-misses-the-point.fr.html>.

<sup>198</sup> LAURENT Philippe, « Logiciels libres: Le droit d'auteur contre le droit d'auteur », op. cit., p. 11.

au principe du logiciel libre, comme nous l'avons vu. Là encore, le code source n'est pas accessible et la modification est interdite.

Enfin, on va également trouver les *cripplewares*, qui sont des *sharewares* munis d'une mesure technique empêchant l'utilisateur de continuer à utiliser le programme au-delà de la période d'essai<sup>199</sup>.

Tous ces logiciels ont un point commun : leur code source n'est pas accessible. Dès lors, on comprend que l'accès au code source prend une place importante dans le concept de logiciel libre.

**72 - Importance de l'accès aux sources.** Comme nous l'avons expliqué le droit d'auteur va protéger la chaîne d'instructions du logiciel, qui est composée du code source et du code objet. Le code objet n'étant pas intelligible par l'homme, il ne présente qu'un intérêt technique pour l'exécution du logiciel.

En revanche, celui qui a accès au code source maîtrise le logiciel : il peut le modifier et ainsi le faire évoluer, corriger les erreurs...

Classiquement, il existe deux façons d'avoir accès au code source : la remise par l'auteur du logiciel et la décompilation.

Dans les contrats de licence portant sur les logiciels, il doit y avoir remise de la chose et de ses accessoires. Néanmoins, la jurisprudence a considéré que le code source ne rentre pas dans la définition des accessoires<sup>200</sup> dans la mesure où, l'utilisateur dispose d'une version en langage machine propre à répondre à ses besoins<sup>201</sup>. Ainsi, sauf cas de négociations particulières, l'auteur n'a aucune obligation de remettre les sources du logiciel.

Concernant la décompilation, le législateur ne l'autorise pas en tant que telle, mais seulement à des fins d'interopérabilité<sup>202</sup>, c'est-à-dire pour assurer « *la capacité*

---

<sup>199</sup> LAURENT Philippe, idem, p. 12.

<sup>200</sup> CA Paris, 11 janvier 1984, jurisdata n°20003.

<sup>201</sup> Excepté pour les contrats portant sur des logiciels spécifiques, lorsque l'auteur du logiciel cède ses droits au client.

<sup>202</sup> Article L 122-6-1 du CPI.



*d'échanger des informations et d'utiliser mutuellement les informations échangées* »<sup>203</sup> entre plusieurs logiciels ou avec du matériel informatique. Ainsi, a été pris en compte le fait que le logiciel est un outil technique et que la disposition du code source peut avoir un intérêt, autre que la contrefaçon, pour l'utilisateur. Mais la décompilation qui est ainsi autorisée reste d'une portée très limitée.

En premier lieu, la décompilation n'est pas chose facile, sauf à disposer d'un logiciel de décompilation, mais surtout, ces exceptions sont soumises à des conditions très restrictives<sup>204</sup>, si bien que l'utilisateur reste relativement tributaire de l'auteur du logiciel. Toutefois, cela démontre que, même en présence d'un logiciel classique, l'accès au code source a un intérêt non négligeable<sup>205</sup>.

Cet intérêt est accru en matière de logiciel libre, car, sans cet accès, les libertés concédées par l'auteur sur son œuvre ne seraient pas effectives. Sans le code source, l'utilisateur ne pourrait pas étudier le fonctionnement du programme, le modifier, l'adapter à ses besoins ou encore redistribuer les modifications.

L'auteur du logiciel autorisant tous les actes d'exploitation sur le logiciel, la décompilation, ici, ne connaît plus les limites posées par la législation : toute personne ayant accepté les termes de la licence peut effectuer cette opération et cela dans n'importe quel but (à condition de respecter les termes de la licence).

On aurait pu se limiter à cette liberté d'accès au code source par le biais de la décompilation. Cependant, la décompilation permet l'accès au code d'assemblage

<sup>203</sup> Préambule de la directive CE n°91/250, 14 mai 1991, JCP 1991, éd. G, III, 64816.

<sup>204</sup> Pour ne citer que celles relatives à la décompilation en vue d'assurer l'interopérabilité : « 1° Ces actes sont accomplis par la personne ayant le droit d'utiliser un exemplaire du logiciel ou pour son compte par une personne habilitée à cette fin ; 2° Les informations nécessaires à l'interopérabilité n'ont pas déjà été rendues facilement et rapidement accessibles aux personnes mentionnées au 1° ci-dessus ; 3° Et ces actes sont limités aux parties du logiciel d'origine nécessaires à cette interopérabilité. Les informations ainsi obtenues ne peuvent être : 1° Ni utilisées à des fins autres que la réalisation de l'interopérabilité du logiciel créé de façon indépendante ; 2° Ni communiquées à des tiers sauf si cela est nécessaire à l'interopérabilité du logiciel créé de façon indépendante ; 3° Ni utilisées pour la mise au point, la production ou la commercialisation d'un logiciel dont l'expression est substantiellement similaire ou pour tout autre acte portant atteinte au droit d'auteur. ». Article L 122-6-1 du CPI.

<sup>205</sup> Cet intérêt est même mis en avant par ceux qui pensent que la protection par droit d'auteur n'est pas forcément la plus adaptée à la création technique qu'est le logiciel, alors qu'avec la protection par brevet, le programmeur serait obligé de divulguer le code source du logiciel. CHALLINE Philippe, « L'informatique et le régime des sources », *Dalloz*, 1997, Chronique n°21, p. 164.

mais pas à la structure originale du code source<sup>206</sup>. Le plus souvent, cela permet d'effectuer une ingénierie inverse pour obtenir un programme similaire mais pas toujours la modification du programme<sup>207</sup>.

C'est pourquoi, avec les logiciels libres comme open source, le code source sera remis par le donneur de licence en même temps que le logiciel<sup>208</sup>. A ce titre, on peut remarquer que certaines licences ne mentionnent pas explicitement l'obligation pour le donneur de licence de communiquer le code source<sup>209</sup>, mais il semble que, pour ce type de licence, on peut considérer le code source comme un accessoire du logiciel, ce qui entraîne pour l'auteur une obligation de le remettre<sup>210</sup>. Non seulement, la nature de ces licences nous amène à cette conclusion, puisque, comme nous venons de le dire, sans cet accès la modification du logiciel sera difficile, mais l'*Open Source Definition*, comme la définition du logiciel libre, exige cet accès. Or, les licences sont sensées se conformer à l'une au moins de ces définitions.

## 2 – Le choix du terme *open source*

73 – De *free software* à *open source*. Comme nous l'avons dit le terme *free* posait problème pour convaincre les entreprises de soutenir le logiciel libre, celles-ci pensant ne pas pouvoir faire de bénéfices avec de tels logiciels. Pour éviter l'amalgame avec la gratuité, certains responsables de la distribution Debian décidèrent qu'il fallait trouver un nouveau terme.

C'est finalement Christine Peterson, du Foresight Institute<sup>211</sup>, qui proposa le terme *open source*, marquant ainsi cette importance de l'accès au code source. Par ce biais, l'objectif des fondateurs de l'Open Source Initiative était aussi de mettre l'emphase

<sup>206</sup> CLEMENT-FONTAINE Mélanie, *La licence publique générale GNU*, op. cit., n°50, p. 30.

<sup>207</sup> Le critère n°2 de l'OSD prévoit d'ailleurs que « *les formes intermédiaires, comme ce qu'engendre [...] un traducteur automatique* » ne sont pas autorisées.

<sup>208</sup> Si le code n'est pas joint au logiciel, il doit au moins être disponible sur demande. Voir critère n°2 de l'OSD : « *Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge* ».

<sup>209</sup> C'est le cas, par exemple, de la licence GPL.

<sup>210</sup> En ce sens : COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 213, p.14.

<sup>211</sup> Site Wikipedia, [http://fr.wikipedia.org/wiki/Open\\_Source\\_Initiative](http://fr.wikipedia.org/wiki/Open_Source_Initiative).

sur la manière dont les logiciels de ce types sont créés et peuvent évolués (l'accès au code source reflétant surtout la liberté de modifier le logiciel) et non plus sur la liberté de l'utilisateur, ce qui intéressait davantage les entreprises<sup>212</sup>.

Ainsi, on voit déjà qu'à travers les choix terminologiques des deux mouvements, une différence d'approche existe<sup>213</sup>.

74 – **Notion de sources.** Dans la mesure où il existe deux conceptions de la notion de sources d'un logiciel, il nous faut préciser ce que l'on entend par accès aux sources en matière de logiciel open source.

Il a en effet été dit que la notion de sources, à laquelle fait référence le Code de la Propriété Intellectuelle, ne devait pas se résumer au code source, c'est-à-dire à la liste d'instructions rédigée en langage évolué, mais devait aussi comprendre « *toutes les informations non publiques nécessaires à un professionnel de l'informatique pour modifier ou maintenir le logiciel considéré, sous une forme compréhensible par ledit professionnel* »<sup>214</sup>. Ceci comprend des indications sur le langage de programmation et la description des structures de programmation (algorithmes, formats de fichier ou des bases de données), soit le matériel de conception préparatoire évoqué par la législation. L'idée, derrière cette conception, est qu'il serait difficile d'assurer la maintenance ou de modifier un logiciel seulement à partir du code source car celui-ci ne permet pas de connaître le processus intellectuel suivi par l'auteur<sup>215</sup>.

Néanmoins, il nous semble que la notion de sources en matière de logiciel open source, tout comme de logiciel libre, ne désigne que le code source à proprement parler.

Les définitions des deux mouvements n'utilisent que ce terme et aucune ne fait référence à une éventuelle documentation du code. Il est toujours possible d'élargir la

---

<sup>212</sup> Voir infra n° 93.

<sup>213</sup> Rappelons cependant que le free software est un logiciel à sources ouvertes tout comme le logiciel open source.

<sup>214</sup> CROZE Hervé et SAUNIER Franck, « Logiciels : retour aux sources », JCP éd. G, 1996, I, 3909.

<sup>215</sup> Idem.

notion par une disposition spécifique dans la licence<sup>216</sup>, mais il semble que ce ne soit pas une obligation<sup>217</sup>.

Dès lors, on voit que la liberté de modification du logiciel est, certes, attribuée à tous les utilisateurs de tels programmes, mais ne sera effectivement exercée que par des personnes techniquement qualifiées. Toutefois, l'utilisateur qui n'a pas les compétences requises pourra toujours faire appel à un prestataire extérieur pour modifier le logiciel en fonction de ses besoins, et faire jouer la concurrence, ce qui n'est pas possible avec un logiciel propriétaire dont l'accès au code source est fermé<sup>218</sup>.

On peut remarquer que dans leur définition des sources au sens large, H. Croze et F. Saunier, estime que la documentation est celle nécessaire à un professionnel pour effectuer la maintenance ou la modification du logiciel. Ceci présuppose qu'en l'absence de cette documentation, même un professionnel ne serait pas à même d'effectuer des modifications du logiciel. On pourrait en tirer deux conséquences : soit l'obligation faite par les licences libres/open source est insuffisante à l'effectivité des libertés accordées, soit la définition large des sources doit être revue et prendre en compte le fait que la documentation facilite la tâche du professionnel mais ne lui est pas nécessaire.

Si le choix du terme *open source* semble remédier à l'amalgame de la gratuité, comme le dit Richard Stallman « *toutes les alternatives ont leur propres défauts* »<sup>219</sup>.

---

<sup>216</sup> Par exemple, l'Open Software License v. 3.0, dans son article 3, prévoit : “ *The term « Source Code » means the preferred form of the Original Work for making modifications to it and all available documentation describing how to modify the Original Work.*”.

<sup>217</sup> En ce sens : TELLIER-LONIEWSKI Laurence et REVEL de LAMBERT Florence, op. cit., p. 37.

<sup>218</sup> REVILLARD Diane, DIERMARK, *Livre blanc organisations et logiciels libre*, op. cit., p. 35.

<sup>219</sup> Richard Stallman, « Pourquoi « logiciel libre » est-il meilleur que « open source » », Site de l'organisation GNU, <http://www.gnu.org/philosophy/open-source-misses-the-point.fr.html>.

## Section II – Le problème du terme *open source*

75 - En fait, tout comme le terme *free*, cette nouvelle appellation ne permet pas de saisir immédiatement ce qu'elle recouvre et peut entraîner des confusions. L'accès au code source est nécessaire en matière de logiciel open source mais il n'est pas suffisant (A). L'Open Source Initiative a cherché des solutions pour éviter ces confusions, cependant nous verrons que leur efficacité n'est pas parfaite (B).

### A – L'insuffisance de l'accès au code source

76 - Si l'accès au code source est nécessaire, nous verrons que ce seul accès ne permet pas la qualification de logiciel open source (1), constat que l'on peut faire au travers de quelques exemples (2).

#### 1 – Une condition nécessaire mais insuffisante

77 – **Le principe.** De la même façon que celui qui détient le support de l'œuvre n'a pas forcément acquis la propriété intellectuelle sur celle-ci, celui qui détient le code source ne bénéficie pas obligatoirement des droits d'auteur sur ce logiciel. Comme le remarquent H. Croze et F. Saunier, s'« *il est vrai que l'absence de remise des sources est un indice de l'absence de transfert de propriété du programme, (...) l'inverse ne prouve rien : on peut communiquer à l'utilisateur un exemplaire des sources sans lui céder les droits d'auteur sur le logiciel* »<sup>220</sup>. Cela a d'ailleurs été rappelé par la jurisprudence<sup>221</sup>. De plus, la cession de droits d'auteur ne se présume

---

<sup>220</sup> CROZE Hervé et SAUNIER Franck, op. cit, p. 96. Dans le même sens voir : CHALLINE Philippe, op. cit., p. 163.

<sup>221</sup> CA Montpellier, 2 juillet 1991, cité in Lamy Droit de l'informatique n° 994.

pas : le Code de la Propriété Intellectuelle impose un certain formalisme<sup>222</sup>, qui rend l'écrit recommandé et d'usage pour tout type de contrat. Dès lors, il n'est pas possible de considérer que, si l'auteur met à disposition le code source de son logiciel, il souhaite, tacitement, laisser libre l'utilisateur d'exploiter son œuvre.

Si les licences open source (et libres) ne transfèrent pas la propriété des droits, le principe que nous avons énoncé n'en est pas moins applicable à la mise à disposition des droits qu'effectuent ces licences, puisque comme nous l'avons expliqué, la cession au sens du Code de la Propriété Intellectuelle doit s'entendre au sens large.

En fait, si les auteurs sont assez réticents à remettre les sources de leur création, dans la mesure où cela accroît le risque de contrefaçon, rien ne les empêche de le faire pour faciliter à l'utilisateur la mise en œuvre des exceptions d'analyse et de décompilation prévues par la législation<sup>223</sup>.

La pratique de la mise à disposition du code source n'a d'ailleurs rien d'exceptionnel. On la retrouve notamment dans les contrats qui portent sur des logiciels spécifiques, dans la mesure où le client en aura probablement besoin pour corriger d'éventuelles erreurs. Mais le créateur du logiciel ne cède pas pour autant automatiquement ses droits<sup>224</sup>.

Dès lors, disposer du code source ne permet pas de copier, distribuer, modifier librement le logiciel, ce qui signifie que cet accès au code source ne permet pas automatiquement de qualifier le logiciel d'open source au sens de l'Open Source Initiative.

---

<sup>222</sup> Notamment l'article L 131-3 qui énonce : « *La transmission des droits de l'auteur est subordonnée à la condition que chacun des droits cédés fasse l'objet d'une mention distincte dans l'acte de cession et que le domaine d'exploitation des droits cédés soit délimité quant à son étendue et à sa destination, quant au lieu et quant à la durée.* ».

<sup>223</sup> Les textes prévoient d'ailleurs cette possibilité puisque l'article L 122-6-1 IV 2° du CPI permet la décompilation si « *les informations nécessaires à l'interopérabilité n'ont pas déjà été rendues facilement et rapidement accessibles aux personnes mentionnées au 1°.* ».

<sup>224</sup> CROZE Hervé et SAUNIER Franck, op. cit.

Le code source est donc une condition nécessaire mais insuffisante : le logiciel devra être accompagné d'une licence manifestant la volonté de l'auteur de mettre à disposition ses droits et de permettre ainsi la modification du logiciel<sup>225</sup>.

## 2 – Exemples

78 – **La confusion.** Néanmoins, ce principe n'est pas évident pour des profanes, et c'est ainsi que naissent des confusions à propos des logiciels open source.

Voici un exemple donné par Richard Stallman : « *l'écrivain Neal Stephenson définit l'open source ainsi : « Linux est un logiciel « open source » signifie, simplement, que n'importe qui peut obtenir des copies de ses fichiers sources »* »<sup>226</sup>.

Or, en réalité, il ne s'agit pas que de conférer l'accès au code source, mais de mettre à disposition les droits d'auteur sur le logiciel.

Ainsi, certains n'hésitent pas à utiliser cette terminologie pour bénéficier de la réputation du libre sans pour autant le pratiquer. C'est d'ailleurs ce que la Free Software Foundation reproche, pour l'essentiel, à cette appellation.

Essentiellement, les logiciels qui vont poser problème sont ceux soumis à des licences plus permissives que les licences propriétaires et qui prévoient toutes l'accès au code source, mais à des conditions contraires aux principes de l'Open Source Initiative (et de la Free Software Foundation également).

79 – **Les licences concernées.** Certaines offrent la possibilité d'exécuter, copier, distribuer et modifier le logiciel mais dans un but non lucratif ou contre paiement d'une redevance en cas d'usage commercial<sup>227</sup>. Il s'agit des licences semi-libres. L'idée

<sup>225</sup> L'Open Source Definition commence ainsi: «*Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria: ...*».

<sup>226</sup> Richard Stallman, « Pourquoi « logiciel libre » est-il meilleur que « open source » », Site de l'organisation GNU, <http://www.gnu.org/philosophy/open-source-misses-the-point.fr.html>.

<sup>227</sup> LAURENT Philippe, Logiciels libres: "Le droit d'auteur contre le droit d'auteur", op. cit., p.12.

On peut remarquer qu'il existe des difficultés pour déterminer ce que l'on entend par usage commercial, dont il n'existe pas de véritable définition légale. Voir en ce sens : Conseil supérieur de la propriété littéraire et artistique, op. cit., n°90.

est de permettre à l'utilisateur de bénéficier du travail d'autrui sans pour autant qu'il puisse en retirer un quelconque profit. Ce type de restriction va à l'encontre des principes de l'open source et du libre, qui permettent l'exercice des droits mis à disposition pour tout type d'usage<sup>228</sup>. C'est le cas, par exemple de la *Simple Machines Licence*<sup>229</sup>.

D'autres permettent la modification du code source mais à des conditions particulières, qui obligent l'utilisateur à soumettre la modification et la distribution du code à l'approbation du donneur de licence.

C'est le cas de la licence communautaire de Sun Microsystems, baptisée SCSL<sup>230</sup> (Sun's Community Source Licence), appliquée au langage de programmation Java. Elle est considérée comme n'étant pas *open source* par l'Open Source Initiative car, d'une part, si on modifie le code et qu'on souhaite le redistribuer, il faut demander l'autorisation de Sun, et, d'autre part, si on souhaite effectuer cette redistribution dans un but commercial, il faut payer des droits à Sun, qui en fixe le montant<sup>231</sup>.

Quant à Microsoft, en réponse à la pression exercée par le mouvement open source, la firme a créé les *Shared Source Licenses* (licence à source partagées)<sup>232</sup>. Ces licences ont pour caractéristique de permettre aux licenciés d'étudier le code source. Le directeur des affaires juridiques et des affaires publiques de Microsoft France a lui-même expliqué<sup>233</sup> que Microsoft avait «*des logiciels qui sont distribués sous des licences dites « ouvertes »*». Mais il continu en disant que ces licences «*permettent d'accéder, dans certaines conditions, aux codes sources de ces logiciels, de les modifier et, dans certains cas de les redistribuer*», et en rappelant combien Microsoft reste attaché à la propriété intellectuelle de ses produits.

<sup>228</sup> Critère n°6 de l'OSD: "The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research."

<sup>229</sup> <http://www.gnu.org/licenses/license-list.fr.html#NonFreeSoftwareLicense>.

<sup>230</sup> Sun Microsystems, [http://java.sun.com/j2se/1.5.0/scsl\\_5.0-license.txt](http://java.sun.com/j2se/1.5.0/scsl_5.0-license.txt).

<sup>231</sup> "Case study of a non open source licence : SCSL", [http://eu.conecta.it/paper/Case\\_study\\_non\\_open.html](http://eu.conecta.it/paper/Case_study_non_open.html).

<sup>232</sup> Microsoft Corporation, <http://www.microsoft.com/resources/sharedsource/default.msp>.

<sup>233</sup> MOSSE Marc, « Logiciels et propriété intellectuelle : des relations originales et enrichies de dynamiques croisées », RLDI, mai 2009, n°49.



Ainsi, dans certaines de ces licences, la modification et la redistribution du code sont trop restreintes pour que ces licences soient considérées comme *open source* au sens de l'Open Source Initiative. En effet, les *Microsoft Limited Permissive License* et *Microsoft Limited Community License* ne permettent l'utilisation du code source qu'aux développeurs travaillant sur la plateforme Microsoft, puisque le code, original comme modifié, doit pouvoir s'exécuter sous Windows. Or, encore une fois l'Open Source Initiative n'admet pas qu'il y ait des discriminations dans les licences : le huitième critère de l'*Open Source Definition* précise, en effet, que « *la licence ne doit pas être spécifique à un produit* ». La *Microsoft Reference License*, elle, interdit toute exploitation du code, qui n'est diffusé qu'à titre d'information<sup>234</sup>.

Toutefois, deux des licences proposées par Microsoft ont été considérées comme réellement *open source* par l'Open Source Initiative : la *Microsoft Public License* et la *Microsoft Reciprocal License*<sup>235</sup>.

Dès lors, il faut faire très attention lorsque l'on a à faire à des logiciels qualifiés d'*open source* et regarder si la licence qui accompagne le logiciel permet la libre exécution, copie, distribution et modification du code source délivré.

Toutefois, l'Open Source Initiative a pris conscience de ces difficultés relatives à l'appellation choisie et à donc cherché des solutions pour y remédier.

## B – Les solutions de l'Open Source Initiative contre l'amalgame

80 – La réservation du terme *open source* par son enregistrement en tant que marque aurait pu être une solution, mais, compte tenu des conditions exigées pour l'enregistrement d'un signe distinctif, cela n'était pas possible (1). L'Open Source Initiative s'est donc tournée vers une marque de certification (2).

<sup>234</sup> GARDETTE Marc, « L'OSI a approuvé deux licences *Shared Source Microsoft* », oct. 2007, disponible sur : <http://blogs.technet.com/b/porte25/archive/2007/10/18/l-osi-a-approuv-deux-licences-shared-source-microsoft.aspx>.

<sup>235</sup> Elles sont également considérées comme libres par la FSF. Voir « Les licences commentées », Site de l'organisation GNU, <http://www.gnu.org/licenses/license-list.fr.html>.

## 1 - L'impossible réservation du terme open source

81 – **La marque.** Selon l'article L 711-1 du Code de la Propriété Intellectuelle, « *la marque de fabrique, de commerce ou de service est un signe susceptible de représentation graphique servant à distinguer les produits et les services d'une personne physique ou morale* »<sup>236</sup>.

En déposant le terme *open source* en tant que marque pour désigner les logiciels élaborés conformément aux principes de l'Open Source Initiative, celle-ci aurait pu empêcher les abus par une action en contrefaçon de marque.

Mais, pour être valable la marque doit être distinctive, c'est-à-dire apte à désigner l'origine des produits ou services qu'elle désigne, comme sa définition l'indique. Par conséquent, sont notamment exclues du champ de la protection les marques génériques ou usuelles et les marques descriptives<sup>237</sup>.

Or, d'après l'article L 711-2 b) du Code de la Propriété Intellectuelle, est descriptif « *le signe ou la dénomination pouvant servir à désigner une caractéristique du produit ou du service, et notamment l'espèce, la qualité, la quantité, la destination, la valeur, la provenance géographique, l'époque de production du bien ou de la prestation de service* »<sup>238</sup>. Il s'agit de ce qui est commun à tous les objets semblables et dont on ne peut autoriser l'appropriation par une personne sous peine de priver ses concurrents de la possibilité de définir la composition, les qualités et les caractéristiques de leurs produits.

<sup>236</sup> En droit américain, le *Trademark Act*, de 1946, indique également que : « *The term "trademark" includes any word, name, symbol, or device, or any combination thereof (1) used by a person, or (2) which a person has a bona fide intention to use in commerce and applies to register on the principal register established by this chapter, to identify and distinguish his or her goods, including a unique product, from those manufactured or sold by others and to indicate the source of the goods, even if that source is unknown.* ».

<sup>237</sup> Le droit américain et le droit français ont des conditions de protection très proches, c'est pourquoi on peut s'appuyer sur le droit français pour expliquer cela. Voir MORVANT Nadège, « Protection des marques : le caractère distinctif des marques en droit français et américain », <http://m2bde.u-paris10.fr/blogs/dpi/index.php/post/2009/01/08/Protection-des-marques-%3A-le-caractere-distinctif-des-marques-en-droit-francais-et-americain-par-Nadège-MORVANT>.

<sup>238</sup> On retrouve une définition similaire dans le *Trademark Act*, art. 6-1-III : « *Trademark registration may be obtained except in any of the following cases: where the mark consists solely of a sign indicating, in a common way, the origin, quality, raw materials, efficacy, use, quantity, shape (including the shape of the packaging) or price of the goods, or the method or time of manufacturing, processing or using them.* ».

Le terme *open source* nous semble être un signe descriptif puisqu'il désigne l'une des caractéristiques essentielles de ce type de logiciels qui est de donner accès au code source<sup>239</sup>.

On peut estimer que même en France le dépôt ne serait pas accepté dans la mesure où ce sont des termes anglais facilement compréhensibles<sup>240</sup>.

Ainsi, la marque ne peut être déposée car non distinctive et il n'y a pas de contrainte légale à son utilisation<sup>241</sup>.

## 2 – Le dépôt de la marque de certification

82 – **La solution ?** L'Open Source Initiative a donc utilisé un autre moyen : la marque de certification<sup>242</sup>, qui est généralement accordée à des produits respectant des normes définies par une personne morale considérée comme compétente pour certifier les produits ou services concernés, mais qui n'est pas réservée aux membres d'une entité et peut être utilisée par toute personne dont les produits ou services en question respectent ces normes<sup>243</sup>.

La marque déposée par l'Open Source Initiative est « *Open Source Initiative* »<sup>244</sup>.

<sup>239</sup> En ce sens : LAURENT Philippe, Logiciels libres: "Le droit d'auteur contre le droit d'auteur", op. cit., p. 14.

<sup>240</sup> Rappelons qu'en France, il est possible d'échapper à la nullité de la marque en utilisant une langue étrangère. En ce sens : Cour de Versailles, 23 mai 2006, « Soldan/Ferrero », RG 05/00803. Cependant, il faut appliquer la règle posée par la CJCE, dans l'arrêt « Henkel » du 12 février 2004, qui estime que le « caractère distinctif d'une marque au sens de l'article 3, paragraphe 1, sous b), de la directive doit être apprécié par rapport, d'une part, aux produits ou services pour lesquels l'enregistrement de la marque a été demandé, et, d'autre part, à la perception des milieux intéressés, qui sont constitués par les consommateurs desdits produits et services. Il s'agit de la perception présumée d'un consommateur moyen des produits ou services en cause, normalement informé et raisonnablement attentif et avisé ». Voir SCHMIDT SZALEWSKI Joanna et PIERRE Jean-Luc, *Manuel droit de la propriété industrielle*, LexisNexis Litec 4<sup>e</sup> édition 2007, n°492. Or, on peut penser que les personnes qui utilisent le type de logiciel en question peuvent saisir immédiatement le sens du terme *open source*, comme faisant référence à l'accessibilité du code source.

<sup>241</sup> Richard Stallman, « Pourquoi « logiciel libre » est-il meilleur que « open source » », Site de l'organisation GNU, <http://www.gnu.org/philosophy/free-software-for-freedom.fr.html>.

<sup>242</sup> Bruce Perens écrit à ce propos : « j'enregistrai la marque de certification, ce qui est une forme particulière de marque déposée, à appliquer aux produits des autres », [http://www.linux-france.org/article/these/the\\_osd/fr-the\\_open\\_source\\_definition-2.html](http://www.linux-france.org/article/these/the_osd/fr-the_open_source_definition-2.html).

<sup>243</sup> Voir site de l'OMPI, « Les marques de certification », [http://www.wipo.int/sme/fr/ip\\_business/collective\\_marks/certification\\_marks.htm](http://www.wipo.int/sme/fr/ip_business/collective_marks/certification_marks.htm).

<sup>244</sup> Voir Annexe 2.

Elle s'accompagne d'un règlement d'usage<sup>245</sup> qui indique que seuls les logiciels distribués sous une licence approuvée par l'Open Source Initiative peuvent se voir appliquer cette marque de certification. Pour qu'une licence soit approuvée, un processus spécifique a été mis en place. Il s'agit avant tout de vérifier que les dix critères de l'*Open Source Definition*, élaborée à cette fin, soient bien respectés<sup>246</sup>.

On remarque que l'on est proche des pratiques des indications d'origine ou des appellations d'origine<sup>247</sup>. La seule différence est que ces dernières sont soumises au contrôle des pouvoirs publics, alors que la marque de certification « *Open Source Initiative* » est complètement gérée par une association privée. La Free Software Foundation procède également à une approbation des licences, mais n'y appose pas de signes permettant leur identification.

Par le biais de cette marque, il est donc plus facile de savoir si on est réellement en présence d'un logiciel open source : il suffit que la licence ou le logiciel possède le logo de l'Open Source Initiative.

Cependant, si cela réduit les risques de confusion, rien n'empêche les développeurs de continuer à utiliser le terme *open source*, puisqu'il ne s'agit pas de l'élément distinctif et dominant de la marque collective.

De plus, comme le fait remarquer Richard Stallman, « *des entreprises font parfois des annonces qui donnent l'impression qu'un programme est un « logiciel open source » sans vraiment le dire explicitement* »<sup>248</sup>.

A titre d'exemple, il cite, notamment, une annonce de Cygnus Solutions qui est la suivante : « *Cygnus Solutions fait partie des leaders du marché de l'open source et vient juste de lancer deux nouveaux produits sur le marché* », produits qui étaient en fait des logiciels propriétaires.

<sup>245</sup> Disponible sur : <http://www.opensource.org/trademark-guidelines>.

<sup>246</sup> Voir « *Approval process* », Site de l'Open Source Initiative, <http://www.opensource.org/approval>.

<sup>247</sup> Point sur lequel nous rejoignons Arnaud Fouquaut. FOUQUAUT Arnaud, *Les licences libres*, mémoire de Master 2 sous la direction de Monsieur le Professeur Théo Hassler, Strasbourg, CEIPI, 2009, p. 52.

<sup>248</sup> Richard Stallman, « Pourquoi « logiciel libre » est-il meilleur que « open source » », Site de l'organisation GNU, <http://www.gnu.org/philosophy/open-source-misses-the-point.fr.html>.

Par conséquent, il convient de toujours vérifier le type et les termes de la licence qui accompagnent un logiciel, même si ce logiciel est désigné comme *open source*.

83 - On a vu au travers de ce chapitre qu'une partie du mouvement libre avait souhaité mettre en place une nouvelle appellation pour les logiciels libres, le terme *free software* engendrant une confusion avec la gratuité. C'est ainsi qu'en 1998, le terme *open source* est apparu sous l'égide de l'Open Source Initiative. Néanmoins, lorsque l'on a saisi ce que recouvrent réellement ces termes, on constate qu'il désigne bien un même type de logiciel, comme nous l'avions démontré en première partie.

Mais si le trouble jeté par ces deux appellations peut être éclairci, l'opposition entre les deux mouvements du libre n'est pas qu'une question de terminologie. En fait, ces deux termes évoquent des idéologies différentes : l'un met l'accent sur la liberté (*free*), l'autre sur le mode de développement (*open source*). Ce qui n'est pas sans conséquences.

## Chapitre 2 – L’opposition idéologique

84 - Rappelons que l’Open Source Initiative a été créée à l’initiative des responsables de la distribution Debian, dont le souhait était d’amener des entreprises à développer des logiciels libres. Pour cela, il fallait démontrer les qualités techniques et les avantages économiques de ce type de logiciel. Outre la mise en place d’une nouvelle appellation, l’Open Source Initiative s’est également détachée du discours éthique de Richard Stallman, pour mettre en place une approche beaucoup plus pragmatique de ce concept, notamment au travers d’une définition propre du logiciel open source : l’*Open Source Definition*.

Ainsi, on constate qu’il existe un désaccord stratégique entre les deux mouvements, bien qu’ils souhaitent autant l’un que l’autre le développement des logiciels libres (Section I). Ce qui conduit parfois à des désaccords pratiques (Section II).

### Section I – Un désaccord stratégique

85 - Pour le mouvement open source, un logiciel non libre est une solution sous-optimale alors que pour la Free Software Foundation, un logiciel non libre est un problème social<sup>249</sup>.

Pour comprendre la vision et les objectifs de chacun des mouvements, nous allons étudier successivement l’approche éthique de la Free Software Foundation (A) puis l’approche pragmatique de l’Open Source Initiative (B).

---

<sup>249</sup> REVILLARD Diane, DIERMARK, *op. cit.*, p. 20.

## A – L’approche éthique de la Free Software Foundation

86 – La Free Software Foundation est souvent décrite comme la représentante d’un mouvement social (1), mouvement qui aujourd’hui s’étend au-delà du logiciel (2).

### 1 – Un mouvement social

87 – **Un idéal de partage.** Comme nous l’avons remarqué, le mouvement libre ne s’inscrit pas dans la négation du droit d’auteur. L’originalité de ce modèle se trouve dans le fait de mettre en avant l’autre faculté conférée à l’auteur par la propriété littéraire et artistique : celle d’autoriser<sup>250</sup>. L’objectif est de démontrer que l’on favorise bien plus la création et la diffusion des œuvres en accordant des libertés aux utilisateurs, que lorsqu’on se place dans une optique d’appropriation<sup>251</sup>. Il est clair que si le mouvement libre utilise l’instrument juridique qu’est le droit d’auteur pour le développement des logiciels libres, l’objectif, clairement affiché par Richard Stallman, est, à terme, de voir la propriété intellectuelle, telle qu’on la conçoit aujourd’hui<sup>252</sup>, disparaître. C’est pourquoi l’idée de liberté est au cœur des principes de la Free Software Foundation.

En fait, la « culture hacker » considérait, jusque dans les années soixante dix, que les logiciels rentraient dans le cadre de la connaissance scientifique (comme les algorithmes, les théories...), qui par définition n’est pas appropriable, dans la mesure où elle fait partie de ce que l’on considère être le « fond commun de l’humanité »<sup>253</sup>.

<sup>250</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al. *op. cit.*, n°448, p. 264.

<sup>251</sup> Rappelons que le droit d’auteur a été conçu, initialement, pour favoriser la création en permettant aux auteurs de vivre de leurs œuvres.

<sup>252</sup> La protection par le droit d’auteur est utile pour les logiciels libres, tant que ce droit existera, pour éviter un détournement de l’œuvre.

<sup>253</sup> TERE Vaden, « Intellectual property, Open Source and Free Software », 2003, <http://opensource.mit.edu/papers/vaden.pdf>.

Mais lorsque l'on a compris la valeur des logiciels, cela a été remis en cause par la volonté d'appropriation, qui a amené à considérer le logiciel comme une œuvre de l'esprit.

Pour la Free Software Foundation, cela n'est pas justifié. L'intérêt général doit primer sur les intérêts particuliers. Richard Stallman explique que, « *de même qu'il existe la liberté d'expression et la liberté de se réunir, il y a la liberté de partager de l'information d'intérêt général. Ce droit devrait être inaliénable* »<sup>254</sup>. L'idée est que le partage de la connaissance permet le développement de la société et que cette connaissance ne doit pas être réservée aux personnes qui ont les moyens financiers d'y accéder.

De plus, la propriété intellectuelle est l'objet d'enjeux économiques majeurs et beaucoup constatent aujourd'hui que certains acteurs, qui se placent entre l'auteur et l'utilisateur, abusent du pouvoir que leur confère cette propriété. De ce fait, le droit d'auteur ne semble ni répondre aux besoins du public, ni même à ceux des auteurs qui pourtant sont ceux que la législation cherche à protéger. Il semble, en fait, répondre à une pure logique de profits, bien que l'évolution de l'informatique favorise une reproduction et une diffusion des œuvres à moindre coût<sup>255</sup>. C'est ce que dénonce Richard Stallman, qui estime que l'on doit protéger le partage « *contre le droit d'auteur, qui est devenu un système d'oppression, un pouvoir injuste entre les mains des éditeurs* »<sup>256</sup>.

Le mouvement libre est donc bien un mouvement social et politique qui cherche, aux travers de la question des logiciels, à véhiculer un idéal de partage de la création. A ce titre, le mouvement ne s'arrête pas aux logiciels. Si Richard Stallman a

---

<sup>254</sup> LEVIN Robert, « Entretien avec Richard Stallman », 23 juin 1998 disponible sur <http://www.linux-france.org/article/these/interview/rms/SomeNet-19980623/>.

<sup>255</sup> STALLMAN Richard, « Pourquoi les logiciels ne doivent pas avoir de propriétaire », extrait de « Free Software, Free Society : The Selected Essays of Richard Stallman », Gnu Press, ISBN: 1-882114-98-1, <http://www.gnu.org/philosophy/why-free.fr.html>.

<sup>256</sup> MOUNIR Roderic, « Richard Stallman, un humaniste au cœur des logiciels », entrevue du 29 septembre 2009, disponible sur : <http://www.equilibres.ch/spip/spip.php?article237>.



créé la notion de logiciel libre, son discours révèle sa volonté que ce concept s'étende au-delà du logiciel<sup>257</sup>. C'est ce qu'il se passe avec l'apparition des œuvres libres.

## 2 – Un mouvement dépassant le cadre du logiciel

88 – **Le concept d'œuvres libres.** A partir du logiciel libre, une véritable « culture libre » est née. Aujourd'hui, elle s'étend à tout type d'œuvre : musique, cinéma, littérature...

Cette extension de la conception d'un droit d'auteur conçu, non pas pour interdire, mais pour autoriser l'utilisation, la copie, la distribution, la modification des œuvres à donner jour à de nouvelles licences. Parmi elles, on trouve la Licence Art Libre et la famille des licences Creative Commons, qui sont les plus populaires.

Concernant les Creative Commons, il s'agit d'un projet né aux Etats-Unis en 2001 à l'initiative de Lawrence Lessig, dont l'objectif est de transformer les « *œuvres protégées par le droit d'auteur en commons, ou biens communs, c'est-à-dire en contenus qui pourrait être utilisé par tous, utilisateurs finaux ou nouveaux créateurs* »<sup>258</sup>. En fait, il existe plusieurs licences Creative Commons accordant des libertés plus ou moins importantes à l'utilisateur en fonction de ce que souhaite l'auteur. La plus restrictive d'entre elles est la licence « Paternité – Pas d'utilisation commerciale – Pas de modification », qui comme son nom l'indique interdit toute modification de l'œuvre ainsi que toute utilisation commerciale, et qui oblige à faire figurer le nom de l'auteur original lors de toute redistribution<sup>259</sup>.

<sup>257</sup> WILLIAMS Sam, STALLMAN Richard, MASUTTI Christophe, *Richard Stallman et la révolution du logiciel libre*, Eyrolles, 2010, p.99. On le voit également dans la définition du logiciel libre où, dans une partie intitulée « Au-delà du logiciel », il est écrit : « *Tout type d'œuvre peut être libre, et la définition du logiciel libre a été étendue à la définition des œuvres culturelles libres applicable à tout type d'œuvre* ». Voir <http://www.gnu.org/philosophy/free-sw.fr.html>.

<sup>258</sup> DUSSOLIER Séverine, « Les licences creative commons : les outils du maître à l'assaut de la maison du maître », *Propriétés intellectuelles*, janv. 2006, n°18, p. 11.

<sup>259</sup> Site Créative Commons, <http://creativecommons.org/licenses/by-nc-nd/2.0/fr/legalcode>.

Si Richard Stallman a lui-même expliqué que, les libertés concédées par l'auteur dépendent du type d'œuvre<sup>260</sup>, avec cette licence, on paraît bien éloigné de sa vision. Il semble donc que seules les licences n'interdisant pas la modification et l'utilisation à des fins commerciales reflètent l'idéal de Richard Stallman : favoriser le partage et l'accès à la culture<sup>261</sup>.

La Licence Art Libre<sup>262</sup>, qui est une initiative française, est peut être celle qui reflète le plus l'idéologie de la Free Software Foundation. En effet, elle accorde, sur les œuvres dont elle définit les conditions d'utilisation, les quatre libertés prévues dans la définition du logiciel libre sans restrictions sur un éventuel usage commercial. Mais surtout, il s'agit d'une licence copyleftée, similaire à la licence GPL, puisque toute distribution de l'œuvre, originale ou modifiée, doit être effectuée sous cette même licence<sup>263</sup>. Ainsi, la liberté de l'utilisateur à l'égard de l'œuvre, mais aussi la pérennité de cette liberté, est garantie.

Il faut, toutefois, remarquer que la transposition du système des licences libres à d'autres types d'œuvres n'est pas sans poser de difficultés en droit français, notamment au regard du droit moral, qui a une place plus importante qu'aux Etats-Unis. Rappelons qu'en droit français, le droit moral de l'auteur est inaliénable, imprescriptible et insaisissable, ce qui signifie que l'auteur ne peut y renoncer<sup>264</sup>.

---

<sup>260</sup> Il classe en trois catégories les œuvres soumises au droit d'auteur : les œuvres fonctionnelles (logiciels, dictionnaires...), les œuvres qui ont un rôle de témoignage (écrits historiques) et les œuvres d'expression personnelle (autobiographies, journaux intimes...). Pour les deux derniers types d'œuvres, il estime que les droits accordés aux utilisateurs doivent être modulés selon le souhait de l'auteur. WILLIAMS Sam, STALLMAN Richard, MASUTTI Christophe, *op. cit.*, p.99.

<sup>261</sup> La FSF ne considère que deux des licences Creative Commons comme libres : la licence « Paternité » et la licence « Paternité – Partage des conditions initiales à l'identique », cette dernière étant une licence copyleftée. Voir « Les licences commentées », Site de l'organisation GNU, <http://www.gnu.org/licenses/license-list.fr.html>

<sup>262</sup> Signalons que cette licence a vocation à s'appliquer à tout type d'œuvre, y compris aux logiciels.

<sup>263</sup> Site Artlibre, <http://artlibre.org/licence/lal>.

<sup>264</sup> Cela s'explique par le fait qu'en France, on considère qu'il y a un lien particulier entre l'auteur et son œuvre. Aux Etats-Unis, il y a une conception plus économique du statut d'auteur, qui fait que le droit moral est réduit au droit à la paternité et au respect de l'intégrité de l'œuvre, et que ce droit n'a pas la même force qu'en France. Voir sur ce point A. STROWEL, « Droit d'auteur et copyright, Divergences et convergences », LGDJ, 1993.

Comme nous l'avons signalé, le logiciel est soumis à un régime spécial, notamment au regard de ce droit moral qui est amoindri. Les licences de logiciels libres ne posaient donc que peu de difficultés au regard de cette question du droit moral.

Mais il en va autrement pour les œuvres libres, pour lesquelles l'auteur bénéficie, outre du droit de paternité, d'un droit de divulgation, de retrait et de repentir, et de respect de l'intégrité de l'œuvre.

Quid alors de l'auteur qui offre la liberté de modifier son œuvre mais décide, ensuite, d'avoir un droit de regard sur ces modifications. S'il ne peut renoncer à son droit au respect de l'intégrité de l'œuvre, il pourra toujours l'invoquer, même si la licence accorde un droit de modification, ce qui crée de l'insécurité juridique pour les licenciés. Nous avons déjà fait ce constat pour les logiciels, mais, ici, l'insécurité est plus grande puisque les atteintes possibles à ce droit ne se limitent pas à celles qui affectent l'honneur ou la réputation de l'auteur.

Quid, également, de l'auteur qui souhaite exercer son droit de retrait ou de repentir. Compte tenu de la large diffusion de l'œuvre, cet exercice, s'il n'est pas prohibé par les licences, apparaît bien théorique, ce qui pose un problème de validité juridique des licences<sup>265</sup>.

Malgré ces interrogations, il semble que ces licences aient bel et bien intégré le paysage juridique français<sup>266</sup> et que le mouvement du libre ne doit plus se résumer aux logiciels. Mais il s'agit là, essentiellement, d'une prolongation de la culture libre et non open source.

**89 – Un mouvement centré sur la liberté.** La définition du logiciel libre donnée par la Free Software Foundation peut être étendue à d'autre type d'œuvre<sup>267</sup>. En effet, cette dernière ne fait pas de l'accès au code source une condition centrale

---

<sup>265</sup> Sur ces questions et, plus généralement, la validité des licences d'œuvres libres voir : SARDAIN Frédéric, « La création contributive sur internet », *RLDI*, nov. 2008, n° 43 / Conseil supérieur de la propriété littéraire et artistique, op. cit., p. 28-30 / CLEMENT-FONTAINE Mélanie, *Les œuvres libres*, op. cit.

<sup>266</sup> Conseil supérieur de la propriété littéraire et artistique, *idem*, p.11.

<sup>267</sup> Voir la définition des œuvres culturelles libres proposée sur le site de la FSF : <http://freedomdefined.org/Definition>.

mais accessoire. Ce qui compte, ce sont les quatre libertés, et l'accès au code source est mentionné uniquement parce qu'il s'agit d'une condition technique nécessaire à l'exercice de ces libertés<sup>268</sup>.

Au contraire, le mouvement open source est centré sur la manière dont sont développés les logiciels et donc sur l'accès au code source. Or, le code source est une notion étrangère aux autres œuvres.

Qu'est ce que le code source d'une musique ? La partition ? Non, car si elle indique les notes jouées, celles-ci peuvent l'être de manières différentes selon le musicien, alors que le code source permet de reproduire un logiciel à l'identique.

Qu'est ce que le code source d'un dessin ou d'une peinture ? Cette question pourrait se poser pour toutes les œuvres non-logicielles et recevrait la même réponse : il n'y a pas d'équivalent<sup>269</sup>, la « source » étant finalement l'œuvre en elle-même.

Dès lors, l'*Open Source Definition* est, à notre sens, difficilement transposable aux licences portant sur ces œuvres, ce qui se justifie car là n'est pas l'objectif de l'Open Source Initiative<sup>270</sup>. De plus, si l'attrait pour les œuvres libres réside également dans le fait de pouvoir les faire évoluer, nous estimons que les enjeux ne sont pas les mêmes. En matière d'art, il n'y a pas vraiment d'avantage technique dans l'ouverture à la modification. On se situe plus dans une sorte de nouveau concept artistique. Une musique, un livre... n'en sera pas forcément mieux conçu, alors qu'un logiciel peut gagner en fiabilité par cette ouverture.

Ainsi, la terminologie comme la stratégie open source ne semblent pas du tout adaptées aux œuvres libres non logicielles, ce qui montre la difficulté qu'il peut y avoir à appréhender de manière globale la notion d'œuvre libre.

On voit clairement que la Free Software Foundation a une idéologie tournée vers l'utilisateur, vers l'accès à l'information pour tous. L'Open Source Initiative, en

<sup>268</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, p. 8.

<sup>269</sup> Nous rejoignons ainsi l'analyse de : FOUQUAUT Arnaud, *Les licences libres*, op. cit., p. 48.

<sup>270</sup> Toutefois, rien n'est impossible. Par exemple, il semble que l'OSL ait vocation à s'étendre à tout type d'œuvre, même si l'une de ses clauses impose la mise à disposition du code source. En ce sens : JEAN Benjamin, *Petit guide à l'usage des licences libres*, op. cit., p. 7. Mais toujours est-il qu'aucune des licences relatives aux œuvres d'art ne figurent dans la liste des licences approuvées par l'OSI.

revanche, dans sa volonté de séduire les entreprises et de s'éloigner de cet aspect social du concept de logiciel libre, s'adresse plutôt aux contributeurs éventuels.

## **B – L'approche pragmatique de l'Open Source Initiative**

90 – L'Open Source Initiative, quant à elle, cherche à présenter le logiciel open source comme une nouvelle méthodologie de développement (1), reflétant ainsi un idéal technique (2) plus que social.

### **1 – Une méthodologie de développement**

91 – **Des logiciels performants.** Nous l'avons dit, pour l'Open Source Initiative un logiciel non libre est une solution sous-optimale. Le logiciel open source est ainsi présenté comme ayant des avantages économiques<sup>271</sup> et techniques par rapport au logiciel propriétaire. C'est grâce à une telle démarche que ces logiciels connaissent, depuis la fin des années 1990, un grand succès auprès des entreprises.

Si la qualité des logiciels open source n'est pas garantie, car ce n'est pas l'objet des licences, la possibilité de modifier le code source, conférée par ces licences, tend à rendre ces logiciels plus performants, plus fiables, puisque les failles de sécurité ou les erreurs dans le code seront plus facilement identifiées par la communauté des utilisateurs<sup>272</sup>.

En fait, là où les logiciels propriétaires vont être développés par une personne ou un groupe de personnes dans une approche centralisée, les logiciels open source, eux, sont évolutifs et développés grâce aux interventions diverses et variées des utilisateurs. Pour Eric Raymond<sup>273</sup>, les premiers sont conçus « *comme des cathédrales*,

---

<sup>271</sup> Sur cette question voir : REVILLARD Diane, DIERMARK, op. cit., p. 32; et Jean Pierre ARCHAMBAULT, *L'industrie du logiciel à la croisée des chemins ? L'économie du libre*, Médialog, n°42 janv. 2002.

<sup>272</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, op. cit, n°32, p. 20.

<sup>273</sup> RAYMOND Eric, *The Cathedral and the Bazaar*, 2000, traduction française par Sébastien Blondeel : <http://www.linux-france.org/article/these/cathedrale-bazar/cathedrale-bazar.html>.

*soigneusement élaborées par des sorciers isolés ou des petits groupes de mages travaillant à l'écart du monde», les seconds ressemblent plutôt « à un bazar, grouillant de rituels et d'approches différents ».*

92 – **Une œuvre évolutive.** Sans cesse modifié, complété, le logiciel open source est une œuvre d'un genre nouveau, qui, si elle présente des avantages techniques, a posé des difficultés quant à son appréhension par le droit d'auteur. Cela a conduit à le considérer comme « *une chaîne d'œuvres dérivées, dont l'analyse détaillée ne peut être correctement effectuée qu'en accomplissant un fractionnement temporel* »<sup>274</sup>. Ce fractionnement se fait entre l'œuvre primaire et les œuvres dérivées successives.

A l'origine d'un logiciel libre, on va toujours trouver un logiciel, qui s'il n'incorpore pas d'œuvre préexistante, sera qualifié d'œuvre primaire. Cette œuvre peut être le fruit d'une seule personne ou de plusieurs développeurs. Dans ce dernier cas, il s'agira d'une œuvre de collaboration ou d'une œuvre collective selon les circonstances<sup>275</sup>.

Pour déterminer où se termine l'œuvre primaire et où commencent les œuvres dérivées, il faut savoir à quel moment l'œuvre primaire peut être considérée comme achevée, sachant que cela ne signifie pas que l'œuvre soit complète compte tenu de la

<sup>274</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al. *op. cit.*, n°88, p. 49.

Contra VIVANT Michel et BRUGUIERE Jean-Michel, Précis droit privé, *Droit d'auteur*, 1<sup>ère</sup> éd., Dalloz 2009, n° 361, p. 261, qui envisage une qualification unique de l'œuvre dans sa globalité en œuvre de collaboration, corroborant les propos de Mélanie Clément-Fontaine qui estime qu'il ne faut pas se cantonner à la vision traditionnelle de la collaboration mais privilégier l'existence chez les intervenants d'une communauté d'inspiration.

Toutefois, une telle vision suppose que les utilisateurs contribuent au logiciel avec un même objectif, ce qui n'est pas forcément le cas, pensons à celui qui le fait pour répondre avant tout à ses propres besoins. De plus, toute modification ultérieure ne pourrait, alors, être considérée comme un apport effectué dans le cadre d'une collaboration, puisque, en matière de collaboration, les droits patrimoniaux doivent être exercés d'un commun accord entre les auteurs, le régime de l'indivision étant applicable. Or, le(s) auteur(s) initiaux ont déjà choisi la licence applicable et, par conséquent, exercé lesdits droits.

<sup>275</sup> L'œuvre de collaboration est « *l'œuvre à la création de laquelle ont concouru plusieurs personnes physiques* », et l'œuvre collective est « *l'œuvre créée sur l'initiative d'une personne physique ou morale qui l'édite, la publie et la divulgue sous sa direction et son nom et dans laquelle la contribution personnelle des divers auteurs participant à son élaboration se fond dans l'ensemble en vue duquel elle est conçue, sans qu'il soit possible d'attribuer à chacun d'eux un droit distinct sur l'ensemble réalisé* », selon l'article L113-2 du CPI. Cette distinction aura pour intérêt de déterminer le titulaire des droits sur l'œuvre.

particularité de ces logiciels. Il semble qu'une fois le logiciel communiqué au public accompagné d'une licence, on puisse le considérer comme étant achevé, dans la mesure où les droits patrimoniaux sont exercés<sup>276</sup>.

Par la suite, tout ajout ou modification apporté au logiciel, si l'apport est original, sera considéré comme une œuvre dérivée ou composite<sup>277</sup>. Là encore, si cette œuvre est le fait de plusieurs personnes, elle pourra être qualifiée de collective ou de collaboration selon les cas.

A partir de ces œuvres dérivées, d'autres pourront être élaborées si l'auteur de l'œuvre primaire et les auteurs des œuvres dérivées ont donné leur accord, et ainsi de suite. C'est ainsi que Philippe Laurent qualifie le développement des logiciels open source de « *chaîne d'œuvres dérivées* ». Mais, il remarque ensuite que cette présentation est loin de représenter la réalité, puisque, à partir d'une œuvre dérivée, plusieurs œuvres dérivées différentes peuvent être développées. Par conséquent, l'évolution du logiciel ressemble plus à un « *arbre* » qu'à une chaîne. Enfin, il recourt à la métaphore de « *la forêt dont les branches des arbres s'entrecroisent* », puisqu'il est possible de créer une œuvre dérivée à partir de lignes de code provenant de logiciels différents (qu'ils soient des œuvres primaires ou dérivées)<sup>278</sup>.

C'est ce système là, ce foisonnement de contributions, que l'Open Source Initiative cherche à mettre en avant.

---

<sup>276</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n°91, p. 50. Toutefois, Mélanie Clément-Fontaine semble considérer que l'œuvre n'est jamais achevée dans le cas des logiciels libres, c'est pourquoi d'ailleurs elle opte pour une qualification globale. Voir CLEMENT-FONTAINE Mélanie, *Les œuvres libres*, op. cit., p. 195.

<sup>277</sup> Article L113-2 du CPI : « *Est dite composite l'œuvre nouvelle à laquelle est incorporée une œuvre préexistante sans la collaboration de l'auteur de cette dernière.* ». Précisons que l'autorisation de l'auteur de l'œuvre préexistante est requise, ce qui est bien le cas avec les logiciels open source grâce à la licence. Le droit français la distingue de l'œuvre dérivée, qui elle correspond à une modification de l'œuvre originaire (comme une traduction par exemple), mais le régime juridique est similaire.

<sup>278</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 96 et suivants, p. 53.

## 2 – Un idéal technique

93 – **La contribution : un point central.** En effet, l'Open Source Initiative conçoit le licencié avant tout comme un éventuel contributeur au logiciel et non comme un utilisateur. D'ailleurs, l'*Open Source Definition* n'évoque pas la liberté d'utilisation du logiciel. Certes, cette liberté semble découler des critères posés, mais elle n'est pas expressément requise.

Pour l'essentiel, les critères tournent autour de l'accès au code source et de la possibilité de le modifier et de distribuer les modifications. La disponibilité du code source est, dans la conception de l'Open Source Initiative, centrale car nécessaire à l'évolution du logiciel. Il s'agit de présenter le logiciel open source comme une nouvelle méthode de développement.

Or, les entreprises ne décident pas de faire des logiciels open source par pur altruisme vis-à-vis des consommateurs, elles souhaitent en retirer certains avantages. En envisageant le licencié comme un contributeur potentiel, l'entreprise y voit le moyen d'améliorer la qualité de ses produits ou encore d'étendre leur champ d'application à moindre coût, en réutilisant le logiciel modifié. C'est aussi un bon moyen de proposer des produits qui correspondent aux besoins réels des consommateurs.

Les contributeurs, eux, ne sont pas lésés, puisque, outre le fait de disposer d'un logiciel qu'ils peuvent adapter selon leurs envies et sans payer de droits d'auteur, ils pourront bénéficier d'une certaine reconnaissance au sein de la communauté.

S'il y a un droit qui n'est pas oublié dans les licences open source, c'est le droit à la paternité de l'œuvre. Dans toutes les licences open source, il existe une obligation relative au droit à la paternité<sup>279</sup> qui permet de garantir l'identification de l'auteur de l'œuvre primaire et des auteurs des diverses contributions, et, ainsi, de protéger leur réputation. C'est ce qui est indiqué dans la justification de l'*Open Source Definition* à

---

<sup>279</sup> Elle se manifeste par des « notices de copyright » mentionnant les divers noms des auteurs.



propos du critère de l'intégrité du code source de l'auteur<sup>280</sup> : « *C'est une bonne chose que d'encourager de nombreuses améliorations, mais les utilisateurs ont le droit de savoir qui est à l'origine du logiciel qu'ils utilisent. Les auteurs et ceux qui font évoluer un logiciel ont aussi le droit, réciproque, de savoir ce qu'on leur demande de faire évoluer, et de protéger leur réputation* »<sup>281</sup>. L'Open Source Initiative va même plus loin, puisqu'elle permet à une licence d'imposer une distribution de fichiers « patch », aux côtés du code source dans le but de modifier le programme au moment de la construction, pour que la distinction entre l'œuvre primaire et l'œuvre dérivée soit plus nette. La Free Software Foundation reconnaît également ce droit de paternité, mais elle privilégie la liberté de l'utilisateur, si bien que cette distribution sous forme de « patch » ne lui convient pas tout à fait<sup>282</sup>.

Toutefois, si cette obligation a un certain intérêt pour les programmeurs, il ne faut pas oublier que le droit de paternité se conçoit aussi bien positivement que négativement : l'auteur a le droit de refuser que son nom figure sur son œuvre. Dès lors, le fait pour certaines licences<sup>283</sup> d'imposer aux contributeurs de mentionner leur nom est assez discutable<sup>284</sup>.

Le seul objectif est donc de créer des logiciels plus performants. On est loin des considérations éthiques de la Free Software Foundation.

94 – « **Having fun** ». Dans ce cadre, l'attrait pour les logiciels open source est purement technique. Il n'y a pas de but politique, pas de lutte pour une cause

---

<sup>280</sup> *The Open Source Definition* : « 4- Intégrité du code source de l'auteur : la licence ne peut restreindre la redistribution du code source sous forme modifiée que si elle autorise la distribution de fichiers « patch » aux côtés du code source dans le but de modifier le programme au moment de la construction. La licence doit explicitement permettre la distribution de logiciel construit à partir du code source modifié. La licence peut exiger que les travaux dérivés portent un nom différent ou un numéro de version distinct de ceux du logiciel original », <http://www.linux-france.org/article/these/osd/fr-osd-1.html>.

<sup>281</sup> « Justification pour la définition de l'« Open Source » », <http://www.linux-france.org/article/these/osd/fr-osd-2.html>.

<sup>282</sup> Voir infra n° 99.

<sup>283</sup> C'est le cas de l'OSL qui énonce : “ *You must cause the Source Code for any Derivative Works that You create to carry a prominent Attribution Notice reasonably calculated to inform recipients that You have modified the Original Work.*”.

<sup>284</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 169, p. 102.

particulière. D'ailleurs, l'Open Source Initiative ne prend pas position sur la question de la propriété intellectuelle. Un logiciel propriétaire ne sera pas forcément considéré comme un « problème » s'il présente une certaine qualité technique. L'Open Source Initiative est beaucoup plus conciliante lorsque quelqu'un souhaite mélanger logiciels libres et logiciels propriétaires, là où la Free Software Foundation tend à encourager l'utilisation du *copyleft* pour éviter de telles pratiques<sup>285</sup>. Pour Linus Torvalds, « *having fun is the ultimate motivation* »<sup>286</sup>. Ce qui compte, c'est la possibilité d'une coopération entre programmeurs, la possibilité de reprendre des codes source existants pour améliorer et accélérer la programmation, puisque, ainsi, ils n'ont pas à repartir de rien à chaque fois qu'ils souhaitent modifier ou améliorer un logiciel<sup>287</sup>.

Cependant, il faut signaler qu'il ne s'agit que d'approches différentes, qui consistent à mettre avant des caractéristiques particulières de ce type de logiciel : un logiciel libre est également un logiciel évolutif et un logiciel open source peut être librement exécuté, copié, distribué et modifié. Néanmoins, il semble que la distinction entraîne quelques différences pratiques.

## Section II – Un désaccord pratique

95 – D'une part, ce désaccord stratégique a conduit les deux mouvements à adopter des définitions qui ne portent pas sur le même objet, ce qui peut avoir des implications pratiques (A). D'autre part, la différence d'idéologie et de définitions amène la Free Software Foundation et l'Open Source Initiative à interpréter les licences qui leur sont soumises de façon différente, ce qui suscite quelques désaccords (B).

---

<sup>285</sup> Voir Chapitre 2, Partie 1.

<sup>286</sup> TERE Vaden, « Intellectual property, Open Source and Free Software », 2003, <http://opensource.mit.edu/papers/vaden.pdf>.

<sup>287</sup> Les logiciels sont des œuvres séquentielles, souvent basées les unes sur les autres. Donner accès au code source, c'est l'intégrer dans un patrimoine commun où chacun peut puiser pour faire progresser l'informatique.

## A – Différences sur l'objet des définitions

96 – **Objets des définitions.** Si on observe les définitions des deux mouvements, on remarque, en effet, qu'elles ne portent pas sur le même objet.

La Free Software Foundation a défini les quatre libertés d'un logiciel libre, la définition porte donc sur le logiciel. L'Open Source Initiative, très pragmatique, a défini dix critères qui permettent de déterminer si une licence est open source ou non, la définition porte donc sur la licence<sup>288</sup>.

Ainsi, un logiciel du domaine public est un logiciel libre, car si le code source est disponible, un tel logiciel répond à la définition de la Free Software Foundation. En revanche, un logiciel du domaine public n'est pas open source car « *les logiciels qui appartiennent vraiment au domaine public ne sont plus protégés par le droit d'auteur et par conséquent on ne peut plus leur appliquer les conditions nécessaires pour qu'ils soient conformes à n'importe quelle licence open source* »<sup>289</sup>. Les utilisateurs sont assurés d'avoir tous les droits que la définition de l'Open Source cherche à assurer parce qu'il n'y a plus de propriétaire pour appliquer de restriction, il n'y a donc pas besoin de licence.

Cependant, tous les logiciels libres ne sont pas des logiciels du domaine public, sinon le concept n'aurait pas d'intérêt. Dès lors, pour les autres logiciels, il faudra également passer par une licence pour obtenir les quatre libertés comme nous l'avons vu plus haut.

Toutefois, cet exemple montre que les deux définitions n'opèrent pas sur le même plan : l'une s'attache au résultat, l'autre au moyen. Or, si la licence est l'outil juridique nécessaire pour atteindre le résultat final qu'est le logiciel libre, elle n'est parfois pas suffisante. Il nous semble alors que le mouvement open source, qui se concentre sur les droits et obligations que la licence doit contenir, puisse passer à côté

<sup>288</sup> JEAN Benjamin, *Petit guide à l'usage des licences libres*, op. cit., p. 3-4.

<sup>289</sup> RADCLIFFE Mark, avocat spécialisé en propriété intellectuelle et conseiller général de l'OSI, in « Le domaine public est-il open source? », de Stephen Shankland, 28 février 2008, <http://www.framablog.org/index.php/post/2008/05/10/domaine-public-et-logiciel-libre>.

de certains points, contrairement à la Free Software Foundation qui envisage plutôt le résultat final.

97 – **La tivoisation.** Il s'agit d'une pratique consistant en la création d'un système qui inclut des logiciels libres, mais utilise le matériel électronique pour interdire aux utilisateurs d'y exécuter des versions modifiées<sup>290</sup>.

Cette expression provient en fait d'un magnétoscope numérique commercialisé par la société TiVo. Ce magnétoscope fonctionne grâce à un logiciel qui intègre le noyau Linux et des logiciels GNU licenciés sous la version 2 de la GPL. Mais ce magnétoscope est équipé d'un système qui vérifie la signature numérique du code exécutable. Si le code a été modifié, le système bloque son exécution et empêche le boîtier de fonctionner. Pourtant, les termes de la licence sont respectés. Le code source du logiciel faisant fonctionner le boîtier est mis à disposition des utilisateurs, et il leur est tout à fait possible de modifier ce code<sup>291</sup>.

Par conséquent, si l'on regarde la licence, celle-ci respecte les critères établis par l'Open Source Initiative, puisqu'elle permet l'accès au code source en vue de sa modification, et les termes de cette licence sont respectés. Comme seule la licence est évaluée par l'Open Source Initiative, on est en présence d'un logiciel open source. En revanche, il ne s'agit pas d'un logiciel libre, au sens de la Free Software Foundation, car les libertés d'utilisation et de modification ne sont pas effectives mais purement théoriques.

La Free Software Foundation a d'ailleurs ajouté la précision suivante à sa définition :  
*« La liberté 1 inclut la liberté d'utiliser votre version modifiée à la place de l'original. Si le programme est livré dans un produit conçu pour exécuter les versions modifiées de quelqu'un d'autre, mais qui refuse d'exécuter les vôtres — une pratique connue sous le nom de « tivoisation » ou (au moyen de listes noires) comme le « secure boot » — la liberté 1 devient une fiction théorique plutôt qu'une pratique de la liberté.*

<sup>290</sup> « Tivoisation », <http://fr.wikipedia.org/wiki/Tivoisation>.

<sup>291</sup> SCHURR Ludovic, op. cit., p. 9-13.

*Ceci n'est pas suffisant. En d'autres termes, ces binaires ne sont pas libres, même si le code source à partir duquel ils ont été compilés l'est* »<sup>292</sup>.

C'est pourquoi, la Free Software Foundation, dans la version 3 de la GPL, a voulu répondre à ce problème grâce à l'article 6, intitulé « *Conveying Non-Source Forms* », qui introduit la notion de « *User Product* » et prévoit que la distribution d'un tel produit, utilisant du code soumis à la GPL v.3, est possible à condition que ce produit ne comporte pas de mécanismes empêchant l'exécution du code objet modifié<sup>293</sup>. L'objectif n'est pas d'interdire toute signature numérique, qui peut avoir un intérêt du point de vue de la sécurité, mais selon Eben Moglen, « *la licence devrait prohiber les moyens techniques d'échapper à ses clauses, avec la même clarté qu'elle prohibe les violations de ces mêmes clauses* »<sup>294</sup>. Ce qui compte, c'est le résultat final, l'effectivité des libertés.

Mais cela conduit à élargir le champ d'application de la licence au matériel sur lequel le logiciel est utilisé, et non plus seulement au logiciel lui-même. Pour les plus pragmatiques, tel que Linus Torvalds<sup>295</sup>, cela ne se justifie pas, d'autant plus qu'il est toujours possible de modifier le logiciel et d'utiliser les modifications sur un autre matériel. On comprend, alors, que ce qui est en jeu n'est pas la possibilité de modifier le logiciel, mais bien la liberté d'utilisation sur le matériel fourni, qui est réduite. Ce

<sup>292</sup> « Définition d'un logiciel libre », Site de l'organisation GNU, <http://www.gnu.org/philosophy/free-sw.fr.html>.

<sup>293</sup> “ *“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made. If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).*”, GPL v. 3, article 6.

<sup>294</sup> “Transcript of Eben Moglen at the 3rd international GPLv3 conference”, 22 juin 2006, <http://fsfe.org/projects/gplv3/barcelona-moglen-transcript.en.html>.

<sup>295</sup> BYFIELD Bruce, « Stallman, Torvalds, Novell et la licence GPL v3 », 29 mars 2007, <http://framablog.org/index.php/post/2007/04/02/Stallman-Torvalds-Novell-et-la-licence-GPL-version-3>.

qui explique, peut être, aussi, pourquoi l'Open Source Initiative ne semble pas s'intéresser à cette question.

98 – **Mise à disposition matérielle.** Dans le même ordre d'idées, comme le remarque Benjamin Jean, la licence n'a que peu d'intérêt si elle ne s'accompagne pas d'une mise à disposition matérielle. Pour lui, « *sans cette licence, l'utilisateur final ne peut exploiter juridiquement l'œuvre ; mais sans une mise à disposition matérielle en conformité avec cette licence, l'utilisateur ne pourra en pratique exploiter le logiciel* »<sup>296</sup>. Ceci signifie que l'on peut très bien avoir un logiciel soumis à une licence open source qui ne soit pas libre, tout simplement parce que les termes de la licence ne sont pas respectés.

La mise à disposition matérielle correspond essentiellement à la mise à disposition d'un code source exploitable. Nous avons dit que l'Open Source Initiative place cet accès au cœur de sa définition, ce qui fait que toute licence open source devra comporter une telle obligation. Mais, selon nous, si le licencié ne respecte pas cette obligation, la licence n'en sera pas moins open source et le seul moyen pour le licencié de pouvoir exploiter le logiciel sera l'action en responsabilité contractuelle. Au contraire, même si la définition du logiciel libre ne considère l'accès au code source que comme une question technique accessoire, tant que les libertés ne peuvent effectivement être exercées par l'utilisateur, il n'y a pas logiciel libre et ce, quand bien même la licence conférerait les droits correspondants.

S'il est certain que cette différence n'entraîne que peu d'incidences pratiques dans la mesure où un auteur a peu d'intérêt à placer son logiciel sous une licence dont il ne respectera pas les termes, l'image positive du libre et de l'open source

---

<sup>296</sup> JEAN Benjamin, *Petit guide à l'usage des licences libres*, op. cit., p. 3.

entraîne, nous l'avons déjà vu, quelques velléités qui peuvent conduire certaines entreprises à adopter ce genre de pratiques<sup>297</sup>.

99 – **Différence de précision.** A l'inverse, le fait pour l'Open Source Initiative de déterminer les critères d'une licence open source lui a permis d'être beaucoup plus précise au regard des droits et obligations que la Free Software Foundation.

Ainsi, la définition de l'Open Source Initiative précise que la licence ne doit pas effectuer de discrimination entre les personnes, les domaines d'application et ne doit pas être spécifique à un produit. Si ces conditions n'apparaissent pas dans les exigences de la Free Software Foundation, elles n'en sont pas moins dans l'esprit du logiciel libre. En effet, la Free Software Foundation précise que « *la liberté d'utiliser un programme est la liberté pour tout type de personne ou d'organisation de l'utiliser pour tout type de système informatique, pour tout type de tâche* »<sup>298</sup>. L'Open Source Initiative précise également que la licence ne doit pas contaminer d'autres logiciels. Là encore, cette disposition est appliquée aux licences libres<sup>299</sup>.

Le critère de distribution est lui aussi pris en compte par la Free Software Foundation. Ce critère nous explique que la licence ne doit pas dépendre d'autres contrats et évoque, en exemple, les accords de non-divulgaration. En fait, cela signifie que de tels accords sont autorisés mais qu'ils ne doivent pas entraver l'exécution de la licence. Or, lorsque l'on regarde les réponses données par la Free Software Foundation aux

---

<sup>297</sup> Idem. On remarque qu'ici le problème est inverse à celui que nous avons évoqué dans la partie sur la confusion suscitée par le terme *open source*. Il ne s'agit plus de donner accès au code source sans conférer les droits nécessaires pour utiliser, copier, distribuer et modifier le logiciel, mais de conférer ces droits sans donner accès au code.

<sup>298</sup> « Définition d'un logiciel libre », Site de l'organisation GNU, <http://www.gnu.org/philosophy/free-sw.fr.html>.

<sup>299</sup> Sur ce point, nous renvoyons à nos développements précédents. Supra n° 56.

questions relatives à l'utilisation de la licence GPL, on constate qu'elle aussi tolère ces contrats tant qu'il ne prive pas d'effet la licence<sup>300</sup>.

Toutefois, l'une des conditions de l'open source fait l'objet d'un désaccord entre les deux organisations. Il s'agit du critère de l'intégrité du code source.

Nous avons dit que l'Open Source Initiative permet qu'une licence oblige le licencié à publier le code source dans sa forme originale à condition qu'elle permette la publication des modifications sous forme de patch. La Free Software Foundation, en revanche, ne semble pas admettre une telle pratique puisqu'elle énonce que « *Si vous publiez vos modifications, vous n'êtes pas obligé de prévenir quelqu'un de particulier ou de le faire d'une manière particulière.* »<sup>301</sup>. D'ailleurs, dans les commentaires sur la *Q Public License* (QPL), qui contient une telle disposition<sup>302</sup>, la Free Software Foundation signale qu'« *elle présente aussi de gros inconvénients pratiques, parce que les sources modifiées ne peuvent être redistribuées que sous forme de correctifs* »<sup>303</sup>. Pour autant, il n'y a pas là de différence profonde entre les deux mouvements, puisque, malgré cela, cette licence est considérée comme libre par la Free Software Foundation.

Mais cette différence de précision permet à la Free Software Foundation d'avoir une plus grande marge d'interprétation à l'égard des licences, ce qui d'ailleurs est l'objet de critiques de la part de ses détracteurs, qui, comme nous l'avons déjà remarqué, estiment que la Free Software Foundation se sert de son autorité pour

---

<sup>300</sup> « **La GPL me permet-elle de distribuer une version modifiée ou beta sous accord de non divulgation ?** Non. La GPL dit que quiconque reçoit une copie de votre version de votre part a le droit de redistribuer des copies (modifiées ou non) de cette version. Elle ne vous donne pas le droit de distribuer votre travail avec plus de restrictions. »,

« **La GPL me permet-elle de développer une version modifiée sous accord de non divulgation ?**

Oui. Par exemple, vous pouvez accepter un contrat pour développer des modifications et accepter de ne pas publier *vos modifications* jusqu'à ce que le client soit d'accord. Ceci est permis car dans ce cas aucun code couvert par la GPL n'est distribué sous accord de non divulgation ».

« FAQ sur les licences GNU », Site de l'organisation GNU, <http://www.gnu.org/licenses/gpl-faq.fr.html#DoesTheGPLAllowNDA>.

<sup>301</sup> « Définition d'un logiciel libre », Site de l'organisation GNU, <http://www.gnu.org/philosophy/free-sw.fr.html>.

<sup>302</sup> Article 3: "You may make modifications to the Software and distribute your modifications, in a form that is separate from the Software, such as patches", QPL v. 1.0, <http://doc.trolltech.com/3.0/license.html>.

<sup>303</sup> « Les licences commentées », Site de l'organisation GNU, <http://www.gnu.org/licenses/license-list.fr.html#GPLIncompatibleLicenses>.



favoriser la licence GPL<sup>304</sup>. Surtout, ce manque de précision quant aux dispositions que doit ou peut contenir la licence rend le concept plus difficile à appréhender car il faut sans cesse estimer si une clause ne contrevient pas à l'une des quatre libertés.

## B – Divergences sur certaines licences

100 – **Interprétation des licences.** Nous l'avons vu, les deux organisations se voient soumettre les licences en vue de leur approbation au regard des définitions établies. Cependant, lorsqu'elles approuvent une licence, elles ne se contentent pas toujours de déterminer si les critères sont remplis, elles interprètent également l'esprit de la licence, qui doit être en accord avec les objectifs qu'elles poursuivent. C'est ce qui est énoncé dans la définition du logiciel libre : « *Pour décider si une licence de logiciel particulière est définie comme libre, nous la jugeons sur ces critères pour déterminer si elle convient à leur esprit tout comme à leur formulation précise. Si une licence inclut des restrictions inacceptables, nous la rejetons même si nous n'avons pas anticipé le problème dans ces critères* »<sup>305</sup>.

Dès lors, certains désaccords sur les licences ont pu apparaître : l'Open Source Initiative acceptant certaines conditions que la Free Software Foundation rejette définitivement.

Trois exemples existent : l'*Apple Public Source License*, la *Reciprocal Public License* et la *Licence Artistic 1.0*.

101 – **La licence d'Apple.** La première version de l'APSL (1.0) comportait trois défauts qui ont amenés la Free Software Foundation à la considérer comme une

<sup>304</sup> COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 3, p. 3.

<sup>305</sup> « Définition d'un logiciel libre », Site de l'organisation GNU, <http://www.gnu.org/philosophy/free-sw.fr.html>.

licence non libre, car chacun de ces défauts était « *suffisant à rendre un logiciel moins que libre* »<sup>306</sup>.

Le premier défaut, intitulé « *Non respect de la vie privée* » par la Free Software Foundation, était que la licence obligeait les utilisateurs à publier leurs modifications, même si elles étaient effectuées et utilisées à titre privé. Cela a conduit la Free Software Foundation à ajouter, dans sa définition du logiciel libre, la précision suivante : « *Vous devez aussi avoir la liberté de faire des modifications et de les utiliser à titre privé dans votre travail ou vos loisirs, sans en mentionner l'existence* »<sup>307</sup>. Il s'agit là d'une atteinte à la liberté de l'utilisateur, dans la mesure où il ne peut pas modifier et exécuter le programme sans en faire profiter la communauté. Or, dans la vision de la Free Software Foundation, pour laquelle la liberté de l'utilisateur prime, même si cette obligation profite à la communauté et permet de faire évoluer le logiciel, elle est intolérable. Il n'est pas dans l'esprit des logiciels libres d'obliger l'utilisateur à redistribuer le programme, modifié ou non. Les obligations du licencié ne doivent prendre effet que s'il décide de publier le programme.

En revanche, cette obligation ne nous semble pas aller à l'encontre des principes de l'Open Source Initiative, puisque celle-ci cherche à favoriser l'évolution du logiciel, en encourageant la contribution des licenciés. Il est même plutôt favorable que toute contribution soit publiée et profite à tous<sup>308</sup>.

Ce premier défaut allait de pair avec le second, intitulé « *Contrôle central* » par la Free Software Foundation, qui consistait à demander à toute personne qui rendait public ou utilisait, autrement que pour de la recherche ou du développement, une version modifiée, de mettre au courant une organisation spécifique, en l'occurrence

<sup>306</sup> « Les problèmes avec les anciennes versions de la Apple Public Source License », Site de l'organisation GNU, <http://www.gnu.org/philosophy/historical-apsl.fr.html>.

<sup>307</sup> « Définition d'un logiciel libre », idem.

<sup>308</sup> Cette obligation de publier les modifications pourrait toutefois être contraire au droit français, puisque cela va à l'encontre du droit de divulgation reconnu aux auteurs, qui leur offre la liberté de décider de communiquer, ou non, leur œuvre au public. Le seul cas où une telle clause serait valable est celui où les modifications ne seraient pas suffisamment originales pour conférer des droits de propriété intellectuelle à leur auteur. En ce sens : COOL Yorick, DE PATOUL Fabrice, DE ROY David, et al, *op. cit.*, n° 163, p. 97.

Apple. Là encore, la Free Software Foundation, pour éviter ce genre de pratique, a ajouté à sa définition la mention suivante : « *Si vous publiez vos modifications, vous n'êtes pas obligé de prévenir quelqu'un de particulier ou de le faire d'une manière particulière* ». Pour les mêmes raisons que précédemment, cette disposition contrevient aux principes de la Free Software Foundation. Mais elle ne satisfait pas non plus ceux de l'Open Source Initiative<sup>309</sup>.

Enfin, le dernier défaut, intitulé « *Possibilité d'annulation à tout moment* », était une clause de résiliation qui spécifiait qu'Apple pouvait annuler la licence et interdire l'utilisation de tout ou partie du logiciel, au cas où quelqu'un formulerait une accusation de violation de brevet ou de *copyright*. Ainsi, la précision suivante est apparue dans la définition du logiciel libre : « *Pour que ces libertés soient réelles, elles doivent être permanentes et irrévocables tant que vous n'avez rien fait de mal ; si le développeur du logiciel a le droit de révoquer la licence ou de changer rétroactivement ses termes, sans que vous n'avez fait quoi que ce soit de mal pour le justifier, le logiciel n'est pas libre* ». Pour la Free Software Foundation, cette clause est néfaste pour les utilisateurs car ils deviennent alors tributaires du système de brevet américain, ce qui n'est pas justifié, notamment pour les utilisateurs résidants dans des pays où la brevetabilité des logiciels est exclue<sup>310</sup>.

Cette clause semble aussi avoir posé problème à l'Open Source Initiative<sup>311</sup>, qui n'a pas approuvé cette première version. En effet, Bruce Perens considère que c'est une

<sup>309</sup> “Section 2.2(c) of the APSL requires that the producer of modifications to APSL-licensed code use a particular URL in the Apple.com domain to notify Apple. While the demise of Apple Computer, Inc. is unlikely in the near future, that sad event would leave us unable to comply with this section of the APSL. **This would constitute a restriction on all rights granted by the license, including those rights necessary to qualify under the Open Source Definition.** The Free Software community plans a very long lifetime for its software, and we hope that Apple will cooperate by changing this provision so that APSL-licensed software could survive without Apple. We suggest that the simple publication of modifications, such as posting on a personal web site accessible to the global internet and pointed out in any binary distributions, be all that is required. This is consistent with other licenses in our community”, PERENS Bruce, “The Apple Public Source License - Our Concerns”, disponible sur: <http://perens.com/Articles/APSL.html>.

<sup>310</sup> « Les problèmes avec les anciennes versions de la Apple Public Source License », Site de l'organisation GNU.

<sup>311</sup> “Section 9.1 of the APSL allows Apple to terminate our rights to use any or all APSL-covered code, at its sole discretion, in the event of an unproven claim of infringement, no matter how specious. This is derived from a similar objectionable portion of IBM's Jikes license, which disqualified that license from being referred to as "Open Source". ”, PERENS Bruce, “The Apple Public Source License - Our Concerns”, disponible sur: <http://perens.com/Articles/APSL.html>.

terminaison arbitraire, qui mettrait en péril les investissements faits par les contributeurs qui auraient apporté des modifications au programme. Pour l'Open Source Initiative une telle clause devrait être limitée à la partie du code supposée représenter une contrefaçon<sup>312</sup>. Cependant, on voit que la Free Software Foundation rejette cette clause en considération de la situation dans laquelle elle place l'utilisateur, alors que l'Open Source Initiative prend en compte les « investissements » des contributeurs.

Par la suite, Apple a modifié sa licence. Dans la version 1.1, ils avaient changé la clause de terminaison en clause de suspension, mais cela ne satisfaisait ni la Free Software Foundation, ni l'Open Source Initiative. Dans la version 1.2, il restait une obligation ne permettant pas de qualifier la licence de libre : toute version modifiée du logiciel « déployée » dans une organisation devait être publiée. En revanche, la clause de terminaison ayant été modifiée et celle relative à la notification de la publication à Apple ayant disparue, l'Open Source Initiative a approuvé cette version. C'est finalement avec la version 2.0 qu'Apple et la Free Software Foundation ont trouvé un terrain d'entente. Dans cette version, « *la définition de « Externally Deployed » (déployé extérieurement) a été réduite de façon appropriée pour respecter les libertés des utilisateurs* » ; ce qui permet à la Free Software Foundation de rappeler que « *cela a toujours été la position de la FSF, que la liberté du logiciel libre soit d'abord pour les utilisateurs de ce logiciel* »<sup>313</sup>. Désormais, cette licence est considérée open source et libre.

Mais ce n'est pas le cas pour les deux autres licences évoquées.

---

<sup>312</sup> “If termination due to an infringement claim is to be allowed at all, it should be explicitly limited to the particular source-code lines that are considered to infringe upon an existing patent”, PERENS Bruce, idem.

<sup>313</sup> « Opinion de la FSF sur la Apple Public Source License (APSL) 2.0 », Site de l'organisation GNU, <http://www.gnu.org/philosophy/apsl.fr.html>.

102 – La *Reciprocal Public License*<sup>314</sup>. Cette licence est considérée comme non libre par la Free Software Foundation pour trois raisons. Tout d’abord, tout comme l’ASPL (dans ses anciennes versions), elle nécessite la publication de toute version modifiée qu’utilise une organisation, même si cette organisation l’utilise à des fins privées (usage interne), et la publication d’une version modifiée nécessite une notification au développeur d’origine<sup>315</sup>.

Ensuite, elle limite le prix qu’un licencié peut demander pour transmettre une copie du logiciel. Or, l’un des principes posés par la Free Software Foundation est la liberté de distribuer des copies pour tout type d’usage, commercial ou non. Limiter le prix que l’on peut demander, c’est restreindre cette liberté. Seul le principe de l’offre et de la demande doit pouvoir intervenir dans cette transaction<sup>316</sup>.

Ainsi, cette licence pose problème car elle donne une conception très restrictive de la sphère privée, tout comme l’ASPL. Seules la recherche ou l’utilisation personnelle du logiciel ne sont pas soumises aux conditions que l’on vient d’énoncer<sup>317</sup>. Elle est néanmoins considérée comme open source par l’Open Source Initiative. Ceci peut s’expliquer de la même façon que pour l’ASPL : il s’agit de limiter les « abus » de personnes profitant du système libre pour faire des bénéfices, sans faire profiter la communauté d’éventuelles contributions. La Free Software Foundation ne cautionne pas forcément ce genre de pratique mais, encore une fois, la priorité est la liberté des utilisateurs.

---

<sup>314</sup> « Reciprocal Public License 1.5 », Site de l’Open Source Initiative, <http://www.opensource.org/licenses/rpl1.5.txt>.

<sup>315</sup> « Les licences commentées », Site de l’organisation GNU, <http://www.gnu.org/licenses/license-list.fr.html>.

<sup>316</sup> « Vendre des logiciels libres », Site de l’organisation GNU, <http://www.gnu.org/philosophy/selling.fr.html>.

<sup>317</sup> L’article 1.2 de la licence stipule: “*Deploy means to use, Serve, sublicense or distribute Licensed Software other than for Your internal Research and/or Personal Use, and includes without limitation, any and all internal use or distribution of Licensed Software within Your business or organization other than for Research and/or Personal Use, as well as direct or indirect sublicensing or distribution of Licensed Software by You to any third party in any form or manner.*”.

103 – La *License Artistic 1.0*. Pour la Free Software Foundation, « *les termes de cette licence sont trop vagues pour que nous puissions parler de logiciel libre; certains passages sont un peu tordus et leur sens est peu clair* »<sup>318</sup>.

L'Open Source Initiative, quant à elle, a approuvé cette licence. Elle a même modifié sa définition pour la prendre en compte<sup>319</sup>, en ajoutant au critère de libre distribution la formule « *en tant que composant d'une distribution d'un ensemble contenant des programmes de diverses origines* »<sup>320</sup>; cette licence prévoyant, notamment, qu'il n'est pas possible de vendre un « *package* » soumis à ses termes, sauf s'il est distribué dans un agrégat, aux côtés d'autres programmes, en tant qu'élément d'une distribution logicielle plus importante<sup>321</sup>. Ainsi, il semble que l'Open Source Initiative admette que l'on puisse interdire la distribution commerciale, à condition qu'elle soit, en revanche, autorisée dans le cadre d'une compilation.

Cependant, il faut constater que cette licence a perdu de son intérêt, puisque les logiciels qui l'utilisent, pour le langage Perl, sont aujourd'hui également sous licence GNU GPL. Cette juxtaposition porte le nom de licence Perl.

L'Open Source Initiative paraît donc plus conciliante que la Free Software Foundation.

On voit que les licences qui ne remplissent pas, à la fois les critères de la Free Software Foundation, et ceux de l'Open Source Initiative, sont peu nombreuses. Cela confirme que les outils juridiques, qui permettent d'obtenir des logiciels libres et open source, sont, dans l'ensemble les mêmes<sup>322</sup>.

<sup>318</sup> « Les licences commentées », Site de l'organisation GNU, <http://www.gnu.org/licenses/license-list.fr.html>.

<sup>319</sup> « Analyse de la définition de l'Open Source », Site de l'Open Source Initiative, <http://www.velic.com/publications/tribunelibre/fr-x2548.html>.

<sup>320</sup> Article 1 de l' OSD: “Free Redistribution. The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources”.

<sup>321</sup> Article 5 de la licence: “You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own.”, disponible sur le site de l'Open Source Initiative, <http://www.opensource.org/licenses/artistic-license-1.0.php>.

<sup>322</sup> Voir Annexe 3.

Cependant, il en ressort que logiciel libre et open source ne peuvent pas totalement être confondus. Un logiciel peut être open source mais pas libre, si les conditions de la licence à laquelle il est soumis sont trop restrictives pour les utilisateurs au sens de la Free Software Foundation.

## Conclusion

104 – Au terme de cette étude, on constate qu'il n'existe pas de différences fondamentales entre un logiciel libre et un logiciel open source d'un point de vue juridique. Une fois dissipées les confusions autour des termes *free* et *open source*, il s'avère que l'un et l'autre désignent des logiciels protégés par le droit d'auteur, que leur auteur permet d'exécuter, copier, distribuer et modifier librement par le biais d'une licence. Il n'y a donc pas de différence au regard du régime juridique qui leur est applicable. De même, les licences qui vont permettre d'atteindre un tel résultat peuvent, dans la majorité des cas, être qualifiées de libre et d'open source.

En réalité, la différence entre libre et open source réside dans les objectifs poursuivis par ces mouvements, objectifs qui n'ont, en pratique, que peu d'incidences juridiques. En effet, même les licences copyleftées, qui pourtant incarnent l'idéologie du mouvement libre, peuvent être qualifiées d'open source.

105 – Cela explique l'émergence du terme FLOSS, qui signifie Free/Libre Open Source Software, et qui tend à rapprocher les deux mouvements. Beaucoup considèrent, en effet, que la division entre la Free Software Foundation et l'Open Source Initiative est contre-productive et ne fait qu'accentuer les confusions autour de ce concept.

Toutefois, la scission entre les deux mouvements persiste depuis plus de dix ans et ne semble pas être amenée à disparaître, compte tenu des fondements idéologiques de celle-ci. De plus, libre et open source ne peuvent pas totalement être confondus, puisqu'il existe malgré tout quelques différences, aussi minimes soient elles. L'adoption du terme FLOSS ne résout donc pas la difficulté que met en avant cette scission du mouvement, c'est-à-dire l'absence d'unification sur la définition du libre et sur l'approbation des licences.

Il nous semble que l'on pourrait alors adopter une approche complémentaire des deux définitions, approche rendue possible du fait qu'elles ne portent pas sur le



même objet. Ainsi, il faudrait vérifier que les critères posés par l'Open Source Initiative sont remplis par la licence et, en même temps, que ces critères conduisent, *in fine*, à conférer les quatre libertés énoncées dans la définition du logiciel libre.

106 - Cependant, il ne faut pas négliger le fait que ces définitions ont été posées par des associations privées qui poursuivent des objectifs précis et s'autorégulent. Dès lors, le processus d'approbation des licences se trouve affecté par l'idéologie, les ambitions de ces associations. L'approche complémentaire peut donc s'avérer inutile. De plus, elle ne constituerait qu'un pis-aller, car il faut également garder à l'esprit que l'extension du mouvement aux œuvres non logicielles perturbe quelque peu la notion<sup>323</sup> et que d'autres définitions ont été proposées<sup>324</sup>.

Tout cela nous incite à penser qu'une harmonisation est nécessaire pour permettre aux utilisateurs, comme aux auteurs, de mieux appréhender la notion d'œuvre libre et surtout, de mieux identifier l'étendue de leurs engagements.

107 – Une telle harmonisation ne serait possible que du fait d'une intervention législative, instaurant une définition à minima. Or, malgré les incitations de la doctrine, le législateur reste le grand absent du monde du libre. En France, hormis l'article L 122-7-1 du Code de la propriété intellectuelle, qui reconnaît la possibilité pour l'auteur de mettre son œuvre à disposition gratuitement, les récentes interventions opérées par les lois DADVSI et HADOPI n'ont rien apporté à la question des œuvres libres.

Il est vrai qu'une définition légale, exempte de toute implication politique ou marketing, serait bénéfique, à la fois pour disposer d'un repère stable et pour entériner une certaine reconnaissance de la notion d'œuvre libre.

---

<sup>323</sup> On pense au critère de mise à disposition du code source qui, en présence de telles œuvres, ne se justifie plus.

<sup>324</sup> Par exemple, celle de l'association Veni Vedi Libri. Voir site de l'association :

[http://venividilibri.org/index.php?title=Licence\\_libre](http://venividilibri.org/index.php?title=Licence_libre)

Ou encore celle de l'association Gulliver. Voir statuts de l'association :

<http://gulliver.eu.org/docs/statuts/statuts.html>

Mais, l'absence d'intervention du législateur peut, en partie, s'expliquer par la difficulté qu'il y a à établir une définition qui fasse l'unanimité.

Là encore, les implications idéologiques des diverses organisations de promotion du logiciel libre et des œuvres libres risque de constituer un obstacle. De même, le législateur français ne peut intervenir seul. Une telle définition devra être internationalement reconnue pour être efficace.

108 - Il faut également prendre en compte le régime juridique qui pourrait être attaché à cette définition. Mélanie Clément-Fontaine propose, par exemple, de mettre en place un régime dérogatoire reposant sur les principes de la propriété collective<sup>325</sup>. Si un tel régime apporterait effectivement un certain équilibre vis-à-vis des principes qui gouvernent la protection des œuvres par le droit d'auteur, il nous semble qu'une partie de la philosophie du libre serait alors mise de côté.

Tout d'abord, parce que pour cet auteur, la réciprocité est une condition qui devrait s'imposer dans un tel régime<sup>326</sup>. Or, nous l'avons vu, les logiciels libres ne peuvent être réduits aux logiciels copyleftés.

Ensuite, parce que, pour nous, un tel système manquerait de souplesse. Les licences offrent l'avantage de pouvoir introduire des dispositions diverses (garanties, licences de brevet...) et de permettre aux auteurs d'ajuster le degré de liberté offert. Il n'est pas évident que les auteurs acceptent d'abandonner leurs droits sans pouvoir réellement maîtriser les conditions de cet abandon. De plus, la liberté ne réside-t-elle pas dans la possibilité de choisir ?

Enfin, si la multiplicité des licences tend à « *asphyxier le système* »<sup>327</sup>, des solutions existent. On pense notamment à l'initiative de l'Open Source Initiative, qui a mis en place un processus de régulation pour éviter les « doublons »<sup>328</sup>, mais aussi à l'étude effectuée par Benjamin Jean sur la compatibilité des licences<sup>329</sup>.

---

<sup>325</sup> CLEMENT-FONTAINE Mélanie, « Faut-il consacrer un statut légal de l'œuvre libre ? », op. cit., p. 75.

<sup>326</sup> Idem, p. 72.

<sup>327</sup> Idem, p. 74.

<sup>328</sup> Voir « The Licence Review Process », Site de l'Open Source Initiative, <http://www.opensource.org/approval>.

<sup>329</sup> JEAN Benjamin, « Option libre : compatibilité entre contrats », op. cit.

Si nous ne rejetons pas en bloc la proposition de Mélanie Clément-Fontaine, force est de constater que le choix d'un régime approprié reste délicat.

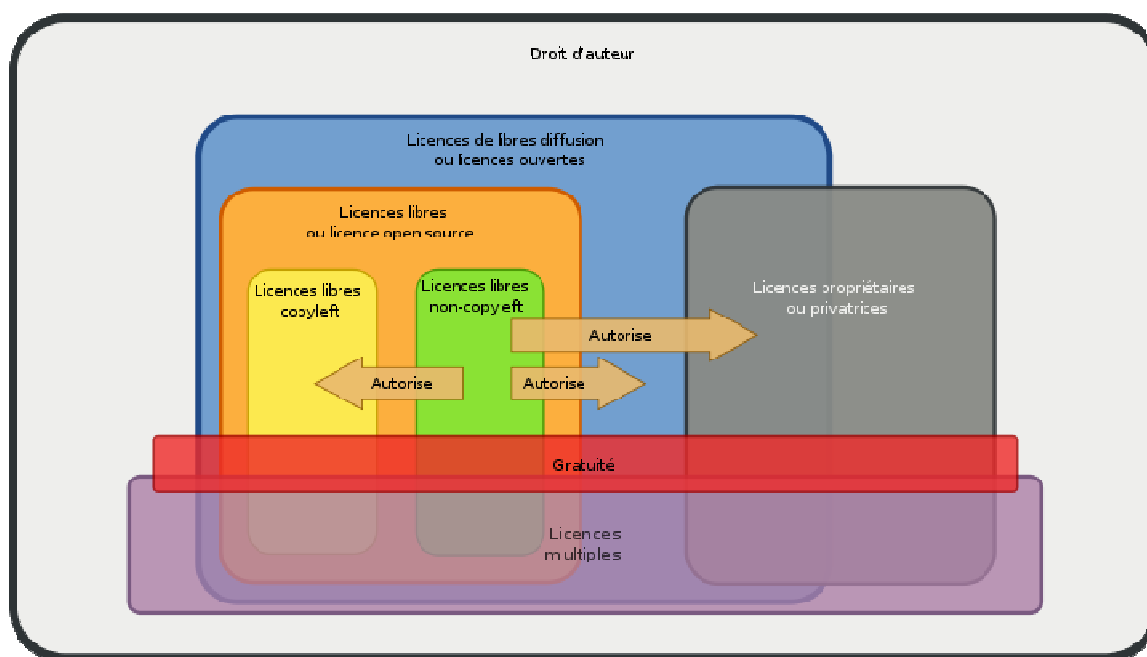
Ainsi, si tous les regards se tournent vers le législateur, l'ampleur de la tâche, mais aussi le contexte actuel<sup>330</sup> rendent la perspective d'une intervention législative plus qu'incertaine...

---

<sup>330</sup> On pense à la « menace » du brevet logiciel qui ressurgit en Europe mais aussi au renforcement de la protection par droit d'auteur avec la loi HADOPI.

## Annexes :

- Annexe 1 : Schéma de classification des licences<sup>331</sup>



- Annexe 2 : Marque « Open Source Initiative »<sup>332</sup>



<sup>331</sup> Schéma proposé sur Wikipédia. [http://fr.wikipedia.org/wiki/Fichier:Classification\\_des\\_licences.svg](http://fr.wikipedia.org/wiki/Fichier:Classification_des_licences.svg).

<sup>332</sup> Voir site de l'Open Source Initiative. <http://www.opensource.org/logo-usage-guidelines>.

▪ Annexe 3 : Tableau des licences de logiciel<sup>333</sup>

	<u>Type</u>	<u>Copyleft</u>	<u>Qualité</u>	<u>System</u>	<u>Famille</u>	<u>Genre</u>
<a href="#"><u>Academic Free License Version 3.0</u></a>	Permissive			Copyright	Rosen	Institutionnelle
<a href="#"><u>Apache License Version 2.0</u></a>	Permissive		Open Source Libre	Copyright		communautaire
<a href="#"><u>Apple Public Source License Version 2.0</u></a>	Copyleft	Standard		Copyright		Institutionnelle
<a href="#"><u>Artistic License</u></a>	Permissive			Copyright		Communautaire
<a href="#"><u>Artistic License Version 2.0</u></a>	Permissive		Open Source	Copyright		communautaire
<a href="#"><u>B.4.JDOM License</u></a>	Permissive		Open Source Libre	Copyright		Communautaire
<a href="#"><u>CLARAty Public License Version 1.0</u></a>	Copyleft	Standard	Non Open Source Non Libre	Copyright		Institutionnelle
<a href="#"><u>CeCILL-B Version 2.0</u></a>	Permissive			Copyright , Droit d'Auteur	CeCILL	Academique

<sup>333</sup> Proposé par le site Veni Vedi Libri. [http://venividilibri.org/index.php?title=Tableau\\_Licences\\_Libres](http://venividilibri.org/index.php?title=Tableau_Licences_Libres).

<a href="#">CeCILL-C Version 2.0</a>	Copyleft	Standard	Libre Open Source	Droit d'Auteur Copyright	CeCILL	
<a href="#">CeCILL Version 2.0</a>	Copyleft	Fort	Open Source Libre	Droit d'Auteur	CeCILL	Institutionnelle
<a href="#">Common Development and Distribution License Version 1.0</a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#">Common Public Attribution License Version 1.0</a>	Copyleft	Fort	Open Source Libre	Copyright		Institutionnelle
<a href="#">Common Public License Version 0.5</a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#">Eclipse Public License Version 1.0</a>	Copyleft	Standard	Open Source Libre	Copyright	Eclipse	Communautaire
<a href="#">Eiffel Forum License Version 1.0</a>	Permissive		Open Source Libre	Copyright		Communautaire
<a href="#">Eiffel Forum License Version 2.0</a>	Permissive		Open Source Libre	Copyright		Communautaire
<a href="#">European Union Public Licence Version 1.0</a>	Copyleft	Fort	Open Source Libre	Copyright , Droit d'Auteur	EUPL	Institutionnelle

<a href="#"><u>Expat License Version 1.0</u></a>	Permissive		Open Source Libre	Copyright		Communautaire
<a href="#"><u>Fair Licence</u></a>	Copyleft	Standard	Open Source Libre	Copyright		Communautaire
<a href="#"><u>GNAT Modified General Public License</u></a>	Copyleft	Fort	Open Source Libre	Copyright	GNU	GNU
<a href="#"><u>GNU Affero General Public License Version 1.0</u></a>	Copyleft	Fort	Non Open Source Libre	Copyright	GNU	GNU
<a href="#"><u>GNU General Public License Version 2.0</u></a>	Copyleft	Fort	Open Source Libre	Copyright	GNU	
<a href="#"><u>GNU Lesser General Public License Version 2.0</u></a>	Copyleft	Standard	Open Source Libre	Copyright	GNU	GNU
<a href="#"><u>GNU Lesser General Public License Version 2.1</u></a>	Copyleft	Standard	Open Source Libre	Copyright	GNU	GNU
<a href="#"><u>GNU Lesser General Public License Version 3.0</u></a>	Copyleft	Standard	Libre Open Source	Copyright	GNU	
<a href="#"><u>Honest Public License Version 1.1</u></a>	Copyleft	Fort	Libre Open Source	Copyright		Communautaire

<a href="#"><u>IBM Public License</u></a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#"><u>Intel Open Source License</u></a>	Copyleft	Standard	Open Source Libre	Copyright	Intel	Institutionnelle
<a href="#"><u>Inter Base Public License</u></a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#"><u>Latex Project Public License Version 1.3c</u></a>	Permissive		Open Source Libre	Copyright		Communautaire
<a href="#"><u>Licence Générale de Traduction</u></a>	Permissive		Non Libre	CopyLeft		Communautaire
<a href="#"><u>Licence KnowledgeTre e Public License</u></a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#"><u>Python Software Foundation License</u></a>	Permissive		Open Source Libre	Copyright		Institutionnelle
<a href="#"><u>Q Public License</u></a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#"><u>RealNetworks Public Source License</u></a>	Copyleft	Fort	Open Source Libre	Copyright		Institutionnelle
<a href="#"><u>Reciprocal Public License</u></a>	Copyleft	Fort	Open Source Non Libre	Copyright		Institutionnelle



<a href="#">Ruby License</a>	Permissive		Open Source Libre	Copyright		Institutionnelle
<a href="#">SPIN License</a>	Copyleft	Standard	Libre ? Open Source ?	Copyright		Institutionnelle
<a href="#">Simple Public License Version 2.0</a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#">Sleepycat Software Product License</a>	Permissive		Libre Open Source	Copyright		Académique
<a href="#">Sun Public License</a>	Copyleft	Standard	Open Source Libre	Copyright	Sun	Institutionnelle
<a href="#">The Swiss Ephemeris Public License</a>	Copyleft	Standard	Non Open Source Non Libre	Copyright		Institutionnelle
<a href="#">True Crypt Collective License Version 1.2</a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#">Verbatim</a>	Verbatim		Non Open Source Non Libre	Copyright		
<a href="#">W3C® Software Notice and License</a>	Permissive		Open Source Libre	Copyright		Académique

<a href="#">X Consortium</a>	Permissive		Open Source Libre	Copyright		Académique
<a href="#">Yahoo! Public License Version 1.0</a>	Copyleft	Standard	Open Source Libre	Copyright		Institutionnelle
<a href="#">Zlib License Version 1.2.2</a>	Permissive		Open Source Libre	Copyright		Académique
<a href="#">Zope Public License Version 2.1</a>	Permissive		Open Source Libre	Copyright		Académique

## Bibliographie :

### ▪ Ouvrages :

- **BERNAULT Carine** et **CLAVIER Jean-Pierre**, *Dictionnaire de droit de la propriété intellectuelle*, Ellipses, 2008, 430 p.
- **BITAN Hubert**, *Droit des créations immatérielles*, Lamy, 2010, 454 p.
- **CARON Christophe**, *Droit d'auteur et droits voisins*, 2<sup>e</sup> éd., Litec 2010, 564 p.
- Code de la propriété intellectuelle, 9<sup>e</sup> éd., Dalloz 2009.
- **CARLUT Christiane**, **ABDI Nidam**, **BANCAL Damien**, et al, *Copyright|Copywrong*; Actes du colloque Le mans\ Nantes\ Saint-Nazaire\ février 2000, éditions Memo 2003, 252 p.
- **COOL Yorick**, **DE PATOUL Fabrice**, **DE ROY David**, et al, *Les logiciels libres face au droit*, Namur, Facultés universitaires Notre-Dame de la paix, Bruxelles, Bruylant 2005, 315 p.
- **GAUTIER Pierre-Yves**, *Propriété littéraire et artistique*, 4<sup>e</sup> éd., Presses universitaires de France, 2001, 835 p.
- **LARROUMET Christian**, Droit civil, *Les obligations – Le contrat*, Tome III 6<sup>e</sup> éd., Economica 2007, 608 p.
- **LE TOURNEAU Philippe**, *Contrats informatiques et électroniques*, 6<sup>e</sup> éd., Dalloz 2010, 438 p.
- **LUCAS André**, **DEVEZE Jean**, **FRAYSSINET Jean**, *Droit de l'informatique et de l'internet*, Thémis droit privé, Presses universitaires de France 2001, 748 p.
- **LUCAS André** et **Henri-Jacques**, *Traité de la propriété littéraire et artistique*, 3<sup>e</sup> éd., Litec 2006, 1210 p.
- **PERLINE** et **NOISETTE Thierry**, *La bataille du logiciel libre : dix clés pour comprendre*, La découverte 2004, 142 p.

- **SCHMIDT SZALEWSKI Joanna** et **PIERRE Jean-Luc**, *Manuel droit de la propriété industrielle*, Paris, LexisNexis Litec 4<sup>e</sup> édition 2007, 696 p.
- **VIVANT Michel** et **BRUGUIERE Jean-Michel**, *Précis droit privé, Droit d'auteur*, 1<sup>ère</sup> éd., Paris, Dalloz 2009, 814 p.
- **WILLIAMS Sam**, **STALLMAN Richard**, **MASUTTI Christophe**, "Richard Stallman et la révolution du logiciel libre", Eyrolles, 2010, disponible sur : <http://framabook.org/stallman.html>.

▪ Articles, chroniques :

- **BELLOIR Philippe**, « L'exclusion de la protection des fonctionnalités d'un logiciel par le droit d'auteur », *RLDI*, 2006, n° 14.
- **BITAN Hubert**, « La loi DADVSI, ou la nécessité de clarifier les notions d'interopérabilité et de mesure technique », *RLDI*, juillet 2007, n°29.
- **CARON Christophe**, « Les licences de logiciels dits « libre » à l'épreuve du droit d'auteur français », *Dalloz*, 2003, chron., p. 1556.
- **CHALLINE Philippe**, « L'informatique et le régime des sources », *Dalloz*, 1997, Chronique n°21, p. 162-166.
- **CLEMENT-FONTAINE Mélanie** :
  - « Les licences creative commons chez les gaulois », *RLDI*, janv. 2005, n° 1.
  - « Faut-il consacrer un statut légal de l'œuvre libre ? », *Propriétés Intellectuelles*, janv. 2008, n°26.
- **CROZE Hervé** et **SAUNIER Franck**, « Logiciels : retour aux sources », *JCP*, éd. G, 1996, I, 3909, n° 8, p. 93.
- **CROZE H.** et **BISMUTH Y.**, « Le contrat dit de licence de logiciel », *JCP E*, 1986, II, 14659, n° 426, p. 121.
- **De ROQUEFEUIL Benoît**, « Le statut juridique des logiciels libres », *Gaz. Pal.*, janv. 2007, n°17, p. 10-13.

- **DUSSOLIER Séverine**, « Les licences creative commons : les outils du maître à l'assaut de la maison du maître », *Propriétés intellectuelles*, janv. 2006, n°18.
- **EGRET Benjamin**, « La nécessaire interprétation des licences relatives aux logiciels libres », *RLDI*, janv. 2006, n°12.
- **GARDETTE Marc**, « L' OSI a approuvé deux licences Shared Source Microsoft », oct. 2007, disponible sur : <http://blogs.technet.com/b/porte25/archive/2007/10/18/l-osi-a-approuv-deux-licences-shared-source-microsoft.aspx>.
- **HUET Jérôme**, « La mise à disposition gratuite d'œuvres sur les réseaux numériques », in *Droit et technique, LexisNexis, Litec*, 2007, p.239.
- **JEAN Benjamin** et **ZAPOLSKY Alexandre**, « La propriété intellectuelle dans l'industrie de l'open source », *Gaz. Pal.*, oct. 2008, n°298-299, p. 19-22.
- **LEVIN Robert**, « Entretien avec Richard Stallman », 23 juin 1998 disponible sur <http://www.linux-france.org/article/these/interview/rms/SomeNet-19980623/>.
- **MOSSE Marc**, « Logiciels et propriété intellectuelle : des relations originales et enrichies de dynamiques croisées », *RLDI*, mai 2009, n°49.
- **MOUNIR Roderic**, « Richard Stallman, un humaniste au cœur des logiciels », entrevue du 29 septembre 2009, disponible sur : <http://www.equilibres.ch/spip/spip.php?article237>.
- **RAMBAUD Sandrine**, « Le juge français et le logiciel libre », *RLDI*, nov. 2009, n°54.
- **SARDAIN Frédéric**, « La création contributive sur internet », *RLDI*, nov. 2008, n° 43.
- **SEDALLIAN Valérie**, « Garantie et responsabilité dans les logiciels libres », *Lamy droit de l'informatique et des réseaux*, n° 152, novembre 2002.
- **SCHURR Ludovic**, « La licence GNU/GPL (V3) a presque deux ans : un bilan mitigé », *Gaz. Pal.*, avril 2009, n° 112-113, p. 9-13.
- **TELLIER-LONIEWSKI Laurence** et **REVEL de LAMBERT Florence**, « Logiciels libres : reconnaissance de la validité de la licence GNU GPL en France », note sous CA Paris 16 septembre 2009, *Gaz. Pal.* 2010, n° 36-37, p. 35-37.

- **THOUMSIN Pierre-Yves**, « Creative commons: le meilleur des deux mondes ? », mai 2008, disponible sur :  
<http://www.droit-technologie.org/upload/dossier/doc/183-1.pdf>.
  - **TRIAILLE Jean-Paul**, « Licences « open source » et contrats avec les développeurs et les distributeurs », *RDTI*, 2005, p. 54-68.
  - **TERE Vaden**, « Intellectual property, Open Source and Free Software », 2003,  
<http://opensource.mit.edu/papers/vaden.pdf>.
  - **VIVANT Michel**, « Logiciel 94 : tout un programme ? », *JCP G*, 1994, I, 3792.
- Mémoires et thèses :
- **CLEMENT-FONTAINE Mélanie** :
    - *La licence publique générale GNU*, mémoire de DEA sous la direction de Monsieur le Professeur Michel VIVANT, Montpellier I/ CNRS, 1999.
    - *Les œuvres libres*, thèse de doctorat, Montpellier I, 2006.
  - **FOUQUAUT Arnaud**, *Les licences libres*, mémoire de Master 2 sous la direction de Monsieur le Professeur Théo Hassler, Strasbourg, CEIPI, 2009.
  - **JEAN Benjamin**, « *Option libre : compatibilité entre contrats* », mémoire de Master 2 sous la direction de Monsieur le Professeur Michel VIVANT, Montpellier I / CNRS, 2006.
  - **LAURENT Philippe**, *Logiciels libres: "Le droit d'auteur contre le droit d'auteur"*, V.A.O. in *intellectuele rechten*, 2003.
  - **LEMYRE Pierre-Paul**, *Les logiciels libres sous l'angle de la responsabilité civile*, mémoire de Maîtrise sous la direction de Monsieur le Professeur Daniel POULIN, Université de Montréal, 2002.
  - **LINSOLAS Julien**, *Le logiciel libre*, mémoire DESS sous la direction de Monsieur le doyen Pierre SIRINELLI, Paris XI - Paris I, 2002.

▪ Etudes, rapports, conférences :

- **Conseil supérieur de la propriété littéraire et artistique (CSPLA)**, *La mise à disposition ouverte des œuvres de l'esprit*, juin 2007, disponible sur : <http://www.cspla.culture.gouv.fr/CONTENU/miseadiposouverterapp.pdf>.
- **JEAN Benjamin**, *Petit guide à l'usage des licences libres*, Matinée juridique du Syntec Informatique du 14 mars 2008.
- **LAFOND Thierry** et **TIAINEN Miia**, *La problématique juridique du logiciel libre*, rapport 2003, disponible sur : [http://www.cndwebzine.hcp.ma/IMG/pdf/La\\_problematique\\_juridique\\_du\\_logiciel\\_libre.pdf](http://www.cndwebzine.hcp.ma/IMG/pdf/La_problematique_juridique_du_logiciel_libre.pdf).
- **LEMYRE Pierre-Paul** et **WILLEMANT Richard**, *Problématiques juridiques des logiciels libres et ouverts : défis et solutions pour le gouvernement du Québec (RESOLL)*, Rapport de projet, Montréal, Avril 2005.
- **REVILLARD Diane**, **DIERMARK**, *Livre blanc organisations et logiciels libres*, Version 1.0, 6 septembre 2005.

▪ Sites internet :

- **Creative Commons** : <http://fr.creativecommons.org/>
- **Framablog** : <http://www.framablog.org/index.php>
- **Free Software Foundation** : <http://fsffrance.org/>
- **GNU** : <http://www.gnu.org/gnu/thegnuproject.fr.html>
- **Jurispédia** : <http://fr.jurispedia.org>
- **Legifrance** : <http://www.legifrance.gouv.fr/>
- **OMPI** : <http://www.wipo.int/>
- **Open Source Initiative** : <http://www.opensource.org/>
- **Veni Vedi Libri** : <http://wiki.vvlibri.org/> et <http://blog.vvlibri.org/>
- **Wikipédia** : <http://fr.wikipedia.org/>

# Table des matières :

<b><i>Introduction</i></b>	<b>5</b>
<b><i>Partie 1 – Logiciel libre et open source : une analogie marquée</i></b>	<b>14</b>
<b>Chapitre 1 – Un concept commun</b>	<b>14</b>
Section I – Une alternative au logiciel propriétaire	15
A – La protection du logiciel par le droit d’auteur	15
1 – Les contours de la protection	16
2 – Le régime de la protection	18
B – Logiciel propriétaire versus logiciel libre/open source	21
1 – Des logiciels soumis au droit d’auteur	21
2 – Une étendue des droits différente	22
Section II – Une possibilité réalisée par le mécanisme contractuel	26
A – La licence comme instrument de permission	26
1 – La faculté d’autorisation	26
2 – Manifestation de volonté par le contrat	27
B – Un instrument valable ?	31
1 – Les interrogations sur la validité	31
2 – La réception par la jurisprudence.	36
<b>Chapitre 2 – Des outils juridiques similaires</b>	<b>37</b>
Section I – La diversité des licences	37
A – Les différentes classifications	38
1 – La classification « historique »	38
2 – La classification analytique	40
B – Licences permissives et licences copyleftées	41
1 – La notion de <i>copyleft</i>	41
2 – Les licences permissives	46
Section II – Une diversité partagée par les deux mouvements	47
A – Le copyleft : un concept rattaché à la Free Software Foundation mais reconnu par l’Open Source Initiative	47
1 – Un concept rattaché au mouvement libre	48
2 – Un concept présent dans le mouvement open source	50
B – Les licences permissives : rattachées à l’Open Source Initiative mais reconnues par la Free Software Foundation	52
1 – Des licences rattachées à l’Open Source Initiative	53
2 – Des licences reconnues par la Free Software Foundation	54



<b>Partie 2 – Logiciel libre et open source : une analogie imparfaite</b>	<b>57</b>
<b>Chapitre 1 – L’opposition terminologique</b>	<b>57</b>
Section I – Le problème du terme <i>free</i>	58
A – L’amalgame de la gratuité	58
1 – Une licence toujours gratuite	58
2 – La distribution à titre onéreux du logiciel	61
B – L’apparition du terme <i>open source</i>	63
1 – L’importance du code source	63
2 – Le choix du terme <i>open source</i>	66
Section II – Le problème du terme <i>open source</i>	69
A – L’insuffisance de l’accès au code source	69
1 – Une condition nécessaire mais insuffisante	69
2 – Exemples	71
B – Les solutions de l’Open Source Initiative contre l’amalgame	73
1 - L’impossible réservation du terme <i>open source</i>	74
2 – Le dépôt de la marque de certification	75
<b>Chapitre 2 – L’opposition idéologique</b>	<b>78</b>
Section I – Un désaccord stratégique	78
A – L’approche éthique de la Free Software Foundation	79
1 – Un mouvement social	79
2 – Un mouvement dépassant le cadre du logiciel	81
B – L’approche pragmatique de l’Open Source Initiative	85
1 – Une méthodologie de développement	85
2 – Un idéal technique	88
Section II – Un désaccord pratique	90
A – Différences sur l’objet des définitions	91
B – Divergences sur certaines licences	97
<b>Conclusion</b>	<b>104</b>
<b>Annexes :</b>	<b>108</b>
<b>Bibliographie :</b>	<b>115</b>
<b>Table des matières :</b>	<b>120</b>

