

## RESEARCH OUTPUTS / RÉSULTATS DE RECHERCHE

### Aspects juridiques des logiciels libres/open source : rapport de recherches sur l'état de l'art

Laurent, Philippe

*Publication date:*  
2009

*Document Version*  
le PDF de l'éditeur

[Link to publication](#)

*Citation for published version (HARVARD):*

Laurent, P 2009, *Aspects juridiques des logiciels libres/open source : rapport de recherches sur l'état de l'art*. CRID, Namur.

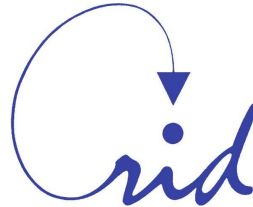
#### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

#### Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



**Aspects juridiques des logiciels libres  
/ open source**  
-  
**Rapport de recherches sur l'état de l'art**  
V. 1.0



Un rapport de :

**Philippe LAURENT (Chercheur Senior – CRID – FUNDP)**

Namur, le 30 septembre 2009

# Table des matières

<b>TABLE DES MATIERES .....</b>	<b>2</b>
<b>I. GENERALITES .....</b>	<b>3</b>
I.1. INTRODUCTION .....	3
I.2. DROIT : QUELQUES NOTIONS DE BASE.....	4
I.3. PETIT GLOSSAIRE TECHNIQUE .....	6
<b>II. PROPRIETE INTELLECTUELLE .....</b>	<b>10</b>
II.1. DROIT D’AUTEUR.....	11
II.1.a. Généralités .....	11
II.1.b. Objet de la protection : l’ « œuvre logicielle ».....	13
II.1.c. Conditions de protection .....	17
II.1.d. Les différents droits des auteurs.....	21
II.1.e. Titularité des droits d’auteur.....	27
II.1.f. Durée des droits d’auteur.....	32
II.2. DROIT DES BREVETS .....	33
II.2.a. Plusieurs systèmes et bases légales multiples.....	33
II.2.b. Objet de la protection : l’invention .....	34
II.2.c. Conditions de brevetabilité.....	37
II.2.d. Formalités .....	38
II.2.e. Droits exclusifs .....	38
II.2.f. Titularité du brevet .....	39
II.2.g. Durée .....	40
II.3. DROIT DES MARQUES .....	41
II.3.a. Plusieurs systèmes et bases légales multiples.....	41
II.3.b. Notions de base.....	42
II.3.c. Droits exclusifs et portée de la marque .....	43
II.3.d. Titularité.....	43
II.3.e. Durée .....	43
II.4. TABLEAU COMPARATIF.....	44
<b>III. ASPECTS CONTRACTUELS .....</b>	<b>45</b>
III.1. LICENCES FOSS : CONTRATS D’AUTORISATION DONNEE SUR LA BASE DE DROITS INTELLECTUELS ....	45
III.1.a. Les licences FOSS sont des contrats .....	45
III.1.b. Règles contractuelles spécifiques en matière de licences de droit d’auteur.....	46
III.1.c. Règles spécifiques en matière de licences de brevets .....	51
III.2. AUTRES ASPECTS CONTRACTUELS .....	52
III.2.a. Obligations du licencié.....	52
III.2.b. Validité du contrat et de certaines clauses .....	55

# I. Généralités

## I.1. Introduction

L'esprit du logiciel libre/open source a toujours existé. Au début de l'ère informatique, les constructeurs d'ordinateurs laissaient l'accès libre aux codes sources des programmes informatiques qui permettaient le bon fonctionnement et l'exploitation de leurs machines. Les constructeurs, avant tout focalisés sur la vente de leurs ordinateurs (hardware), avaient en effet tendance à considérer le logiciel (software) comme étant un accessoire sans véritable valeur économique.

Sous l'effet, entre autres, de la généralisation des outils informatiques, de la standardisation des composants et de la place que prirent les fonctionnalités logicielles au sein du travail journalier des entreprises, cette vision du logiciel changea, les constructeurs de hardware arrêtaient de permettre l'accès aux codes sources de leurs logiciels, et de nouveaux business apparurent autour de l' « édition logicielle ».

Lorsque lui fut refusé l'accès aux codes sources du driver de son imprimante, la frustration de Richard Stallman<sup>1</sup> fut grande et le poussa à créer la FSF (Free Software Foundation). Dans la foulée, il créa la licence « copyleft » GPL afin de s'assurer que ses logiciels soient (et restent) disponibles sous forme de code source et modifiables à volonté... le droit d'auteur fut utilisé, non plus pour limiter les actions des utilisateurs, mais pour assurer la pérennité de « libertés ».

Un juriste aura tendance à faire remonter la genèse des logiciels libres à la création des premières licences libres/open source, c'est-à-dire, à la première approche juridique visant à résoudre le problème factuel de l'accès aux sources. En effet, face à une pratique paraissant à l'époque choquante pour certains, à savoir le refus de fournir les codes sources des logiciels, une solution fut trouvée par le biais d'un usage « subversif » des droits reconnus aux développeurs : le droit d'auteur fut utilisé afin de permettre (et parfois forcer) le libre accès aux codes sources et d'autoriser à tout un chacun de les modifier comme bon lui semble.

Le but du présent rapport est d'analyser les droits et mécanismes juridiques impliqués dans le cadre de la concession de licences libres/open source en Belgique, et de déterminer les questions spécifiques qui méritent de faire l'objet d'études plus spécifiques dans le cadre du projet CELLAVI.

---

<sup>1</sup> Développeur américain à présent célèbre pour son engagement et sa participation au développement du mouvement du logiciel libre.

## I.2. Droit : quelques notions de base

Ces quelques lignes d'introduction sur le mouvement libre/open source illustrent d'emblée un point élément essentiel : il s'agit, dans toute situation, et à chaque stade de l'analyse de clairement **distinguer le fait du droit**.

En matière de licences libres/open source, relèvent entre autre de l'ordre du fait :

- l'accès au code source et ce que cela permet (la possibilité matérielle de modifier les logiciels, de les améliorer, de les combiner, de les déposer en libre accès, de les partager...)
- les fonctionnalités des logiciels, leur qualité, leur usage,...
- les communautés se créant autour des logiciels, leur organisation, leurs pratiques, etc.
- les services commerciaux relatifs aux logiciels libres,...

Le droit concernera avant tout :

- la « propriété intellectuelle » relative à ces logiciels et les droits exclusifs qu'elle implique dans le chef des titulaires de droit,
- les licences accompagnant les logiciels, leur étendue et leurs spécificités (et, entre autres, les autorisations de copier, modifier, distribuer, communiquer les logiciels...)
- les droits et devoirs des donneurs de licences et des licenciés (qui impliquent, par exemple et selon les cas, l'obligation de fournir les codes sources)
- les règles contractuelles applicables à ces licences,...

Traditionnellement, une distinction doit être faite entre le droit subjectif et le droit objectif. Le **droit objectif** peut être défini comme l'ensemble des règles qu'une société donnée s'impose à elle-même. Le **droit subjectif** est un ensemble de prérogatives qui découle du droit objectif et permet aux sujets du droit d'obtenir des choses ou d'interdire certaines choses sur la base du droit objectif. Par exemple, nous aurons l'occasion d'analyser dans le détail les règles de droit d'auteur. Ce dernier (élément du droit objectif) permet aux auteurs de s'opposer à la reproduction de leurs œuvres (droit subjectif).

Le droit objectif peut être subdivisé en plusieurs matières, parmi lesquelles le droit civil et/ou commercial, le droit pénal, et le droit public et/ou administratif. Le **droit**

**civil et commercial** édicte une série de règles régissant le comportement entre individus, entre autres, en déterminant leurs obligations et leurs droits (subjectifs). Le non respect de ces règles peut être sanctionné devant les cours et tribunaux civils, devant lesquels les personnes ayant subi un dommage ou dont les droits n'ont pas été respectés pourront en faire forcer le respect et/ou réclamer réparation. Le **droit pénal** édicte une série d'obligation de comportement en société. En cas de non respect de ces règles, des sanctions (peines) sont prévues par la loi. Les infractions sont recherchées et poursuivies par le ministère public, qui représente les intérêts de la société. Le **droit public ou administratif** constitue l'ensemble des règles relatives au fonctionnement des administrations et de leurs rapports avec les citoyens.

Au sein du droit civil et commercial, on peut distinguer une multitude de branches :

- soit en fonction des instruments législatifs ou leur objet législatif (droit des personnes, droit de la preuve, droit des contrats, droit des hypothèques, droit de la propriété littéraire et artistique, droit des marque, droit des brevets, droit de la responsabilité civile ou contractuelle, etc.)
- soit en fonction de secteurs d'activité (droit de la construction, droit des nouvelles technologies, droit médical, droit bancaire, etc.).

Chaque secteur fait cependant appel à de multiples instruments législatifs ou principes généraux (par exemple, le droit des contrats, le droit de la responsabilité civile, le droit de la preuve, etc.). Les lois ou règles plus spécifiques concerneront davantage un ou plusieurs secteurs spécifiques.

Le secteur qui est avant tout visé en matière de logiciels est celui du droit des nouvelles technologies, ou droit des technologies de l'information et de la communication (droit des TIC). Dans pareil cadre, le droit de la propriété intellectuelle et le droit des contrats prennent une place prépondérante.

Une première section de se rapport sera consacrée aux aspects relatifs à la **propriété intellectuelle** des logiciels libres/open source (voir point II). Une seconde section étudiera les **aspects contractuels** et autres aspects juridiques des licences de logiciels libres/open source (voir point III). Mais avant tout, nous proposons de nous accorder sur quelques termes utilisés dans cette étude et repris sous la forme d'un « petit glossaire technique » (voir point I.3).

## I.3. Petit Glossaire Technique

Ce glossaire a été constitué en ayant pour seul objectif de définir de façon simplifiée certains termes qui seront utilisés dans ce rapport ou dans d'autres analyses spécifiques subséquentes. Il n'est ni exhaustif, ni scientifique. Le vocabulaire qui y est repris n'est pas juridique, ni défini par une quelconque législation, mais peut être considéré comme étant un vocabulaire et des acceptions fréquemment utilisés dans le milieu.

**Application** : logiciel qui permet de réaliser une ou plusieurs fonctions.

**Code objet** : code du logiciel dans sa forme exécutable par une machine, c'est-à-dire en « langage machine ».

**Code source** : code du logiciel dans sa forme modifiable et éditable par l'homme, c'est-à-dire en « langage de programmation ».

**Compilateur** : logiciel permettant de traduire un code source en code objet.

**Compilation** : processus par lequel du code source est traduit en code objet, grâce à un compilateur.

**Driver** (« pilote » / « module ») : logiciel permettant à un autre logiciel (un système d'exploitation) d'interagir avec un périphérique.

**F.S.F.** (Free Software Foundation<sup>2</sup>) : organisme constitué aux Etats-Unis afin de promouvoir les logiciels libres et de défendre les droits des utilisateurs de logiciels libres.

**FOSS** (free/open source software) : acronyme utilisé afin de parler indistinctement des « logiciels libres » et des « logiciels open source ».

**FLOSS** (free/libre/open source software): voir "FOSS".

---

<sup>2</sup> Voy. <http://www.fsf.org>

**Hardware** (matériel informatique): ensemble des équipements électroniques (appareils informatiques) composé par l'ordinateur et ses périphériques.

**Librairie** (terme usuel francophone basé sur une mauvaise traduction du terme anglais « library » - aussi appelé « bibliothèque ») – logiciel qui se présente sous la forme d'une multitude de fonctions regroupées de façon systématique dans le but d'être mises à la disposition d'autres logiciels sans devoir être réécrites au sein de ces-derniers. Il s'agit en d'autres termes d'une sorte de boîte à outils (fonctions) utilisables (appelables) par d'autres logiciels.

**Logiciel** (software) : programme informatique permettant à un système informatique d'assurer certaines fonctions et qui constitue une série d'instructions adressées au « hardware ».

**Logiciel Libre** (free software) : logiciel qui répond à la définition établie par la Free Software Foundation sur la base de quatre libertés indispensables que sa licence doit garantir<sup>3</sup> à savoir :

- la liberté d'exécuter le programme pour n'importe quel usage;
- la liberté d'étudier le fonctionnement du programme et de l'adapter à ses besoins, ce qui implique la nécessité de pouvoir accéder au code source ;
- la liberté de redistribuer des copies ;
- la liberté d'améliorer le programme et de publier ses améliorations, ce qui entraîne également la nécessité d'accéder au code source.

**Logiciel Open Source** (Open Source Software) : logiciel dont la licence remplit les dix critères édictés par l'Open Source Initiative et constituant l'« Open Source Definition ».

**Logiciel Propriétaire** : terme générique employé afin de désigner les logiciels qui ne sont pas des Logiciels Libres ou des Logiciels Open Source. Sont généralement visés par ce terme les logiciels distribués sous des licences payantes et restrictives (à l'inverse des logiciels libres ou open source).

**Open Source Definition** : Définition de ce qu'est un Logiciel Open Source selon l'OSI. Cette définition est constituée de dix critères<sup>4</sup>, qui peuvent être résumés comme suit:

- la licence ne peut pas interdire de vendre ou donner le programme en tant que composant d'un ensemble de programmes (une « distribution »)

<sup>3</sup> Voy. <http://www.gnu.org/philosophy/free-sw.html>; également disponible en français à l'adresse <http://www.gnu.org/philosophy/free-sw.fr.html>

<sup>4</sup> Voy. <http://www.opensource.org/docs/definition.php>

provenant de différentes sources; la licence ne peut prévoir de paiement de royalties pour pareille distribution;

- le programme doit inclure les codes sources, et la licence doit permettre la distribution en code source ou sous forme compilée; si le programme est distribué sous forme compilée, la licence doit prévoir l'accès au code source; Il est interdit de proposer un code source rendu volontairement difficile à comprendre; Il est également interdit de soumettre des formes intermédiaires, comme le résultat d'un préprocesseur ou d'un traducteur automatique.
- la licence doit garantir la liberté de pouvoir créer des versions modifiées et des œuvres dérivées du programme, et de pouvoir distribuer celles-ci sous les mêmes conditions que la licence du logiciel initial;
- la licence doit prévoir la possibilité de distribuer du logiciel issu de la modification des codes sources ; la licence peut cependant assurer l'intégrité du code source initial en limitant la redistribution du code source d'un logiciel modifié à la distribution du code source initial accompagné de « patches » de modifications et/ou en requérant que les logiciels dérivés portent des noms et des numéros de versions distincts de ceux du logiciel initial;
- la licence ne peut pas établir de discrimination entre personnes ou groupements de personnes;
- la licence ne peut restreindre l'usage du logiciel à certains champs d'activités;
- les droits transmis avec le logiciel doivent être retransmis à toute personne à qui le logiciel est redistribué sans qu'une licence complémentaire ne soit nécessaire;
- la licence ne peut être spécifique à un produit; elle ne s'applique qu'au programme concerné sans que cela ne dépende du fait que ce logiciel soit redistribué dans un ensemble spécifique (une « distribution » spécifique);
- les logiciels distribués avec celui soumis à la licence ne doivent pas nécessairement être « open source »;
- les termes de la licence doivent respecter le principe de neutralité technologique.

**Ordinateur** : Usuellement, l'ordinateur peut être décrit comme étant l'ensemble composé de l'unité centrale et les Périphériques principaux : écran, souris, clavier. Les ordinateurs portables comprennent tous ses éléments ou équivalents dans un seul appareil. Le terme Ordinateur est également parfois utilisé dans un sens restrictif, pour désigner uniquement l'Unité Centrale.

**O.S.I.** (Open Source Initiative<sup>5</sup>) : organisme sans but lucratif de droit américain constituée afin d'établir l' « Open Source Definition » et de certifier les licences qui en remplissent les critères.

---

<sup>5</sup> Open Source Definition version 1.9 - Voy. <http://www.opensource.org/docs/definition.php>

**Patch** : partie de logiciel contenant un ensemble de modifications apportées à un logiciel ; il se présente sous la forme d'une série de séquences de modifications, permettant généralement de corriger, mettre à jour ou « upgrader » un logiciel ;

**Périphérique** : partie de « hardware » ajoutée à une Unité Centrale et permettant d'interagir avec celle-ci ou lui ajoutant une fonctionnalité (imprimante, scanner, souris, etc.)

**Système d'Exploitation** (« Operating System » / « OS ») : programme ou ensemble de programmes qui permettent la liaison entre le hardware et les applications informatiques. Le Système d'Exploitation peut être composé de Drivers et a à sa charge l'ordonnancement de l'accès par les programmes aux divers ressources et périphériques de l'ordinateur.

**Unité Centrale** : partie du « hardware » effectuant le traitement des données et exécutant les logiciels.

## II. Propriété intellectuelle

La propriété intellectuelle peut être définie de manière simplifiée comme étant l'ensemble des droits reconnus par certaines lois spécifiques aux hommes qui ont fait preuve d'un travail de création intellectuelle ou d'innovation. Elle se subdivise traditionnellement en deux catégories, à savoir la propriété littéraire et artistique, d'une part, et la propriété industrielle, d'autre part.

La propriété littéraire et artistique comprend les droits d'auteur, ainsi que des droits satellites à ceux des auteurs et appelés « droits voisins » (droits des artistes interprètes, des producteurs de phonogrammes et des organismes de radiodiffusion). On y inclut aussi également le droit des bases de données.

On notera que le droit d'auteur est le droit intellectuel qui fut choisi en tant que protection « normale » des logiciels, et que, dès lors, les licences FOSS sont avant tout des licences de droit d'auteur (voir point II.1)

La propriété industrielle vise davantage les droits concernant directement le commerce et l'industrie. On y classe, entre autres, le droit des brevets, le droit des marques, le droit des dessins et modèles et le droit des obtentions végétales.

Le droit des brevets est de plus en plus abordé dans le cadre des licences FOSS (voir point II.2).

Les marques sont des signes distinctifs bien présents dans le milieu open source et importants dans le cadre des activités commerciales liées au FOSS (voir point II.3).

## II.1. Droit d'auteur

### II.1.a. Généralités

#### II.1.a.1. La protection « naturelle » des logiciels

Le droit d'auteur est la principale protection des logiciels, ceux-ci ayant été assimilés aux « œuvres littéraires » par les principaux traités internationaux sur la propriété intellectuelle<sup>6</sup> ainsi que par la directive européenne harmonisant le droit des états membres de l'union européenne en matière de protection des programmes d'ordinateur<sup>7</sup>. Ces accords internationaux décrétant la protection des logiciels par le droit d'auteur ont permis de faire profiter les développeurs de logiciels d'un système de propriété intellectuelle préexistant, déjà bien établi et largement harmonisé sur le plan international (par la convention de Berne entre autres).

Une première remarque préliminaire s'impose dès lors en matière de logiciels FOSS: ceux-ci sont protégés par le droit d'auteur, non seulement en Belgique, mais également dans tous les pays étant tenus par les traités, accords et/ou directives ci-dessus cités.

Si les licences FOSS sont souvent rédigées selon un droit national déterminé ou, en tous les cas, inspirés par ce-dernier, il s'agit de souligner que, par le fait qu'il s'agit avant tout de licences de droit d'auteur, leur validité dans les autres pays, en tous les cas en ce qui concerne leurs effets généraux, devrait en principe être reconnue.

Malgré tout, l'harmonisation internationale n'étant pas complète, c'est dans l'interprétation et l'application pratique des licences que certaines difficultés pourraient apparaître dans le cadre de situations bien déterminées.

Ce sont donc avant tout des considérations d'ordre pragmatiques qui ont poussé les législateurs à protéger les logiciels par le système de droit d'auteur... Une autre solution aurait été de créer un système spécifique (« *sui generis* ») de protection des logiciels. Cela aurait permis de créer un droit « sur mesure » parfaitement adapté à son objet, mais cette solution aurait impliqué le fait de partir de zéro, non seulement au niveau de la conception de ce droit, mais également en matière d'harmonisation internationale du système de protection.

<sup>6</sup> Voy. par exemple l' art.10 des TRIPs et l'art.4 du WCT.

<sup>7</sup> Voy. art.1 de la directive 91/250/CEE

Si le fait de protéger les logiciels par le droit d'auteur était sans conteste la solution la plus pratique du point de vue de l' « ingénierie juridique », il faut cependant noter qu'il n'était pas totalement adapté à ce nouvel objet de protection. Nous verrons entre autres que certaines règles spécifiques relatives aux logiciels ont été adoptées, et que malgré cela, certaines critiques sont encore adressées envers le système pour son inadéquation à certaines situations techniques.

On notera également déjà que le droit des brevets est perçu par certains acteurs comme étant un second moyen de protéger les logiciels (voir point II.2).

### **II.1.a.2. Protection gratuite, systématique et sans formalité**

L'une des caractéristiques principales et intéressantes du droit d'auteur est que ce droit intellectuel s'acquiert gratuitement, systématiquement et sans que l'auteur n'ait à remplir aucune formalité.

**Systématique** : le droit d'auteur s'acquiert au fur et à mesure que le logiciel se crée. En effet, le droit d'auteur protège immédiatement l'œuvre (le logiciel) au moment où elle prend forme (c'est-à-dire, entre autres, au moment où le code est composé).

**Sans formalité** : il ne faut faire aucune « déclaration » ou « enregistrement » pour obtenir la protection par le droit d'auteur. Le droit d'auteur n'est pas accordé par un organisme, mais est immédiatement acquis par l'effet de la loi.

**Gratuite** : il ne faut pas payer de frais d'enregistrement ou de maintien du droit d'auteur. Il est valide dès la création de l'œuvre et s'éteint 70 ans après la mort de son auteur originaire (personne physique qui a créé le logiciel).

Il s'agit de souligner les conséquences de ces principes en matière de logiciels FOSS. Les logiciels FOSS sont et restent protégés par le droit d'auteur, et ne doivent pas être confondus avec les œuvres du domaine public ou « libres de droit ».

Le domaine public est l'ensemble des œuvres et de tous autres éléments qui ne sont pas (ou plus) protégés par un quelconque droit de propriété intellectuelle. Il est généralement constitué d'œuvres ou éléments non protégeables, ou dont la protection est expirée. Par exemple, les œuvres protégées par le droit d'auteur tombent dans le domaine public 70 ans après la mort de leur auteur.

A l'inverse, les logiciels FOSS sont toujours protégés par le droit d'auteur, raison pour laquelle ils sont accompagnés d'une licence. La différence principale entre les logiciels open source et les logiciels dits « propriétaires » est le degré de permissivité des licences qui les accompagnent.

### **II.1.a.3. Système belge des droits d'auteur protégeant les logiciels**

En Belgique, la protection des logiciels par le droit d'auteur est consacrée par la loi du 30 juin 1994 transposant en droit belge la [directive](#) européenne 91/250/CE du 14 mai 1991 concernant la protection juridique des programmes d'ordinateur ([LPO](#)). Il s'agit d'une loi spécifique séparée de la loi générale relative au droit d'auteur. Entre autres, elle définit certains droits des auteurs en tenant compte de leur objet spécifique, elle prévoit également des exceptions propres à la technologie logicielle, et elle met en place certaines mesures pour lutter contre la contrefaçon.

Cependant, cette loi spéciale ne couvre pas l'entièreté des règles du droit d'auteur relatif aux logiciels et n'est donc pas autonome: elle doit en effet être complétée, le cas échéant, par les dispositions de la loi générale, également du 30 juin 1994, relative au droit d'auteur et aux droits voisins ([LDA](#)).

Dès lors, lorsque la LPO est muette sur certains aspects du régime de droits d'auteur appliqué aux logiciels, c'est dans la LDA que la solution sera recherchée.

On notera que cet aspect dual des lois applicables à la protection des logiciels par les droits d'auteurs n'est pas sans embarras. En effet, certains problèmes épineux ont pour origine la question de savoir si un thème est effectivement traité par la LPO, et si oui, dans quelle mesure il est traité, afin de déterminer si certaines règles de la LDA doivent effectivement s'appliquer ou non en matière de logiciels.

### **II.1.b. Objet de la protection : l'« œuvre logicielle »**

#### **II.1.b.1. L'œuvre dans toutes ses formes**

Afin de leur faire profiter de la protection par le droit d'auteur, les logiciels ont été assimilés aux œuvres littéraires et artistiques. De la même façon que l'auteur d'un livre le crée au fur et à mesure des mots, lignes et paragraphes qu'il rédige, l'auteur

d'un logiciel crée du code qu'il structure et compose afin de composer son « œuvre logicielle ».

On remarquera dans nombreuses licences FOSS que le logiciel est parfois également appelé « work » (à savoir « œuvre » en anglais).

Le logiciel est cependant une œuvre un peu particulière dans la mesure où un même logiciel peut prendre plusieurs formes naturelles en fonction du degré d'abstraction auquel on se situe et/ou du langage dans lequel il est présenté.

En effet, la première étape de création d'un logiciel est tout d'abord, à un niveau d'abstraction élevé, la conception de « *flow-charts* » ou **diagrammes de flux**. Il s'agit généralement d'un schéma de séquences où les flux de données sont généralement représentés par des flèches et les traitements de données par des « boîtes ». Ces représentations permettent de donner un aperçu des processus complexes qui seront implémentés par le logiciel. Cette phase du développement est importante puisqu'elle permet d'avoir une vue d'ensemble du logiciel et d'analyser les interactions entre les différentes parties de ce dernier.

Le logiciel est ensuite codé dans un langage de programmation choisi par le développeur. Il se présente alors sous la forme d'une longue suite d'instructions écrites dans un langage technique, certes, mais compréhensibles par des développeurs maîtrisant le langage en question. Ce code, appelé **code source**, est dès lors compréhensible et modifiable aisément par l'être humain. Parfois, certaines formules ou certaines séquences peuvent également faire l'objet de commentaires, permettant à un développeur autre que le développeur originaire de suivre le raisonnement de ce dernier.

On notera que les licences FOSS définissent souvent ce que l'on entend par code source. La GPLv3 le décrit comme étant *“the preferred form of the work for making modifications to it”*<sup>8</sup>, et l'EURL comme étant « l'Œuvre lisible par l'être humain dans la forme la plus appropriée pour l'examiner et pour y faire des modifications »<sup>9</sup>.

Généralement<sup>10</sup>, le code n'est cependant exécutable par une machine que s'il est compilé en **code objet**, qui lui est en langage « machine » incompréhensible à l'être humain. Les logiciels propriétaires traditionnels et « tout public » sont généralement distribués exclusivement sous cette forme « objet », ce qui empêche toute modification du programme par l'utilisateur.

---

<sup>8</sup> Voy. Art.1 de la GPLv3.

<sup>9</sup> Voy. Art. 1 de l'EURL.

<sup>10</sup> Certains codes dits « interprétés » ne nécessitent pas de compilation mais sont immédiatement exécutables.

Le logiciel est indistinctement protégé par le droit d'auteur dans ces trois formes (ainsi que les formes intermédiaires qui peuvent parfois se présenter). L'article 1 de la LPO prévoit en effet que sont protégés par le droit d'auteur et assimilés aux œuvres littéraires non seulement les programmes d'ordinateur, mais également « *le matériel de conception préparatoire* »<sup>11</sup>. Les programmes d'ordinateur sont également protégés en tant que tels mêmes s'ils sont incorporés au matériel<sup>12</sup>.

Une remarque importante s'impose cependant : ni la LDA, ni la LPO, ni même la directive européenne d'ailleurs, ne donnent une définition de ce qu'est un « programme d'ordinateur ». Cette absence de définition dans la loi s'explique par la volonté de respecter un principe de bonne pratique législative consistant à rester technologiquement neutre. En effet, c'est la crainte que la définition donnée ne devienne immédiatement obsolète qui poussa à n'en reprendre aucune dans la loi. Cependant, selon les travaux préparatoires de la directive (qui datent de 1989) le terme « programme » visait à désigner « l'expression, dans toute forme, tout langage, toute notation ou tout code, d'un ensemble d'instructions ayant pour objet de permettre à un ordinateur d'accomplir une tâche ou une fonction particulière »<sup>13</sup>.

Dans la mesure où les programmes d'ordinateurs sont régis par certaines règles spécifiques qui dérogent au droit d'auteur général, la question de ce qu'est un programme d'ordinateur n'est pas sans incidences pratiques.

Par exemple, un cas de jurisprudence impliqua un employeur qui prétendit qu'un site web était protégé par la LPO afin de tenter de tirer profit de la présomption de cession prévue à l'article 3 de cette loi (*cf. infra*). Le juge chargé de l'affaire<sup>14</sup> releva que l'employeur ne prouvait pas qu'un logiciel propre avait été développé par l'employé afin de créer le site web, et que, quand bien même quelques lignes de programmation auraient pu avoir été écrites afin de créer ce site web, ces lignes ne présentaient pas une originalité suffisante (*cf. infra*) pour attirer la protection spécifique de la LPO.

On ne conclura cependant pas de cette jurisprudence que tous les éléments liés à la constitution de sites webs sont systématiquement exclus de la protection par la LPO. Par exemple, certains contenus ou aspects visuels sont en effet, le résultat de programmations complexes ou obtenus grâce à des scripts qui pourraient, quant à eux, se voir qualifier de programmes d'ordinateur. Dans ce cas, l'aspect visuel du site web sera protégé par le droit d'auteur traditionnel, et les scripts, par la LPO.

---

<sup>11</sup> Au sujet de la protection des diagrammes de flux, voy. par exemple Civ. Bruxelles (cess.), 30 juin 2003, *A&M*, 2004/2, p. 153 et s.

<sup>12</sup> Directive 91/250/CE, considérant 7.

<sup>13</sup> COM (88) 816 final – SYN 183, *J.O.C.E.*, 12 avril 1989, n° C 91/9.

<sup>14</sup> Civ. Gand (cess.), 3 septembre 2001, *I.R.D.I.*, 2002, p. 104 et s., note E. LAEVENS.

## II.1.b.2. La forme et non les idées

Le droit d'auteur protège la forme et non les idées. Ce principe signifie que seule la mise en forme du logiciel est protégé, et non les idées qu'il implémente.

Reprenant mot pour mot la directive 91/250/CEE concernant la protection juridique des programmes d'ordinateurs<sup>15</sup>, la LPO prévoit que « *les idées et principes à la base de tout élément d'un programme d'ordinateur, y compris ceux qui sont à la base de ses interfaces, ne sont pas protégés par le droit d'auteur* ».

La jurisprudence française<sup>16</sup> a eu l'occasion de se prononcer sur ces notions, en énonçant que les fonctionnalités d'un logiciel ne bénéficient pas, en tant que telles, de la protection du droit d'auteur, dès lors qu'elles ne correspondent qu'à une idée. Les fonctionnalités sont définies dans cette jurisprudence comme étant « *la mise en œuvre de la capacité [du logiciel] à effectuer une tâche précise ou à obtenir un résultat déterminé* ».

La tâche consistant à distinguer les fonctionnalités de la forme d'expression est cependant beaucoup plus ardue en pratique, non seulement en droit d'auteur en général, mais encore plus particulièrement dans le cadre du développement logiciel<sup>17</sup>.

En effet, entre la fonctionnalité (idée abstraite non protégée) et le code (résultat concret sous la forme de lignes pouvant défiler à l'écran) s'insèrent une série de concepts à des degrés d'abstraction intermédiaires tels que la structure, les séquences, l'organisation, les routines, sous-routines, etc... qu'il sera toujours difficile de classer dans la logique binaire « protégéable » (car relevant de la mise en forme) / « non protégéable » (car relevant de l'idée).

En pratique, lorsqu'il s'agira de détecter des contrefaçons (à savoir la violation des droits d'auteur sur un logiciel) ce principe donnera lieu à deux types de litiges, à savoir :

- les cas simples de reproduction à l'identique de code (« copy-paste »): le logiciel dans son niveau d'abstraction la plus basse a été reproduit à la virgule près, il s'agit inmanquablement d'une reprise de sa forme ;
- les cas complexes : similarité d'éléments situés à un niveau d'abstraction supérieur (comme la structure, sectionnement, les séquences,...) où il s'agira d'abord de trier ce qui relève de l'idée ou non avant de les comparer avec les éléments similaires du code « original ».

<sup>15</sup> Directive 91/250/CEE du Conseil du 14 mai 1991 concernant la protection juridique des programmes d'ordinateur, *J.O.C.E.*, 15 mai 1991, n° L 122/42.

<sup>16</sup> Cass. fr. (1<sup>ère</sup> ch. civ.), 13 décembre 2005, disponible sur le site <http://www.legalis.net> (dernière visite le 25 février 2009).

<sup>17</sup> Voy. par exemple à cet égard PH. LE TOURNEAU, *Contrats informatiques et électroniques*, Paris, Dalloz, 2006, p. 149 ;

La volonté du législateur est de ne pas permettre à quelqu'un de « monopoliser » une fonctionnalité par le biais du droit d'auteur, et donc de laisser l'opportunité à quiconque de pouvoir utiliser cette fonctionnalité, à condition de la coder dans une forme différente.

Nous avons déjà souligné le fait que seul le code source est compréhensible par l'être humain. On comprendra donc que l'accès au code source des logiciels permettra plus facilement de détecter les reprises littérales, et de « comparer » leurs formes.

Par contre il permettra aussi d'avantage de comprendre le fonctionnement du logiciel et les idées, concepts et algorithmes qui les sous-tendent, et de réutiliser ces éléments non protégeable dans un autre logiciel dont la forme sera, quant à elle, autre et également originale.

## II.1.c. Conditions de protection

Traditionnellement, la protection par le droit d'auteur n'est reconnue à une œuvre que si elle est mise en forme et qu'elle est originale.

### II.1.c.1. Mise en forme

Ce premier critère est intimement lié à l'objet de protection du droit d'auteur et à son étendue<sup>18</sup>. Nous avons en effet expliqué plus haut que seule la forme (ou également « l'expression ») du logiciel était protégée et non les idées et fonctionnalités qu'il met en œuvre.

Le corollaire est qu'il est nécessaire que l'œuvre soit mise en forme avant de pouvoir accéder à la protection par le droit d'auteur. Ainsi, le seul fait de penser à un projet de logiciel, seul ou à plusieurs, n'est pas protégé par le droit d'auteur : il faut en effet qu'il fasse effectivement l'objet d'une réalisation concrète (présentation graphique, encodage, etc.).

Nous avons cependant vu que le logiciel était protégé à divers niveaux d'abstraction, et qu'un diagramme des flux était déjà protégé en tant que tel par le droit d'auteur (*cf. supra*). Cette condition n'implique donc pas forcément que le logiciel soit codé en tant que tel, mais plutôt exprimé de n'importe quelle façon tangible et/ou sensible.

---

<sup>18</sup> A. STROWEL & E. DERCLAYE, *Droit d'auteur et numérique*, Bruxelles, Bruylant, 2001, p. 37.

## II.1.c.2. Originalité

L'originalité<sup>19</sup> est sans doute le concept clé de la protection par le droit d'auteur, dans la mesure où il intervient au moins à deux stades importants de la protection, à savoir au moment de déterminer l'octroi ou non de droits d'auteurs sur une œuvre, ainsi qu'au moment d'apprécier la contrefaçon. En effet, la protection par le droit d'auteur ne sera accordée que sur les parties et/ou aspects originaux de la forme (*cf. supra*) d'un logiciel, et il n'y aura de contrefaçon que lorsque ces parties et/ou aspects ont été repris dans le logiciel contrefaisant. L'exercice consistera dès lors plutôt à distinguer, au sein même du programme, ce qui est original de ce qui ne l'est pas, pour ensuite constater, en cas de litige, si ce sont ces éléments qui ont fait l'objet d'une reprise.

A défaut d'une définition dans la LDA, l'originalité en droit d'auteur est généralement décrite comme étant l'empreinte de la personnalité de l'auteur<sup>20</sup>. Selon la jurisprudence de la Cour de cassation, pour être originale, une œuvre doit être « *marquée par la personnalité de son auteur* »<sup>21</sup>. La Cour de cassation présente également l'originalité comme étant « *l'expression de l'effort intellectuel de celui qui l'a réalisée* »<sup>22</sup>. Ces deux définitions diffèrent en ce que la première repose sur un aspect essentiellement subjectif (la « personnalité de l'auteur ») alors que la seconde met l'accent sur une conception plus objective, à savoir l'évaluation du « travail intellectuel ».<sup>23</sup>

L'article 2 de la LPO précise la notion d'originalité en ce qui concerne les logiciels : « *Un programme d'ordinateur est protégé s'il est original, en ce sens qu'il est une création intellectuelle propre à son auteur. Aucun autre critère ne s'applique pour déterminer s'il peut bénéficier d'une protection par le droit d'auteur* »<sup>24</sup>.

Cette condition de protection, ainsi que sa rédaction, furent reprises telles quelles de l'article 1.3 de la directive.

La question épineuse qui se pose au sujet de ce critère est de savoir si la définition traditionnelle et « subjectiviste » de l'originalité en droit d'auteur, selon laquelle l'originalité se traduit par la présence de la « *marque de la personnalité de l'auteur* »,

<sup>19</sup> Sur cette notion, Voir par exemple. A. STROWEL, « L'originalité en droit d'auteur: un critère à géométrie variable », *J.T.*, 1991, pp. 513-518; F.GOTZEN « Het hof van cassatie en het begrip « oorspronkelijkheid » in het Belgische auteursrecht », *Computerr.*, 1990/4, p. 161.

<sup>20</sup> A. BERENBOOM, *Le nouveau droit d'auteur et les droits voisins*, Bruxelles, Larcier, 1995, p. 48.

<sup>21</sup> Voy par exemple Cass., 25 octobre 1989, *Pas.*, 1990, I, p.239 ; Cass., 27 avril 1989, *Pas.*, 1989, I, p. 908 ; Cass., 2 mars 1993, *Pas.*, 1993, p. 234.

<sup>22</sup> Cass., 27 avril 1989, *Pas.*, 1989, I, p. 908.

<sup>23</sup> Pour plus d'information sur ces notions, voy. par exemple F. DE VISSCHER & B. MICHAUX, *Précis du droit d'auteur*, Bruxelles, Bruylant, 2000, p. 15 et A. STROWEL & E. DERCLAYE, *op. cit.* p. 26.

<sup>24</sup> Cette définition provient de l'article 1<sup>er</sup> al. 3 de la Directive, dont le 8<sup>ème</sup> considérant ajoute encore que « *les critères appliqués pour déterminer si un programme d'ordinateur constitue ou non une œuvre originale ne devraient comprendre aucune évaluation de la qualité ou de la valeur esthétique du programme* ».

reste d'application en matière de logiciels ou si, au contraire, la « *création intellectuelle propre* » doit s'apprécier distinctement, en tenant compte avant tout du « travail intellectuel » de l'auteur.

Le 30 juin 2003, le président du tribunal de première instance de Bruxelles, statuant comme en référé, estima que devait être protégé par le droit d'auteur « *ce qui n'était pas développé par hasard, mais qui était le résultat d'un processus de recherche et développement long et coûteux, et donc, d'une réflexion créative, une création intellectuelle propre à l'auteur dans le sens de la loi sur la protection des programmes d'ordinateur* »<sup>25</sup>. Cette définition retenue par le président semble davantage mettre l'accent sur l'investissement intellectuel du créateur, concrétisé par ses recherches et sa réflexion, et non sur la question de savoir si le code reflète sa personnalité<sup>26</sup>.

Dans une décision du 23 février 2004, la cour d'appel d'Anvers résuma, peu adroitement selon nous, la question comme suit : « *en général, une œuvre jouit de la protection par le droit d'auteur à condition qu'elle soit l'expression de l'effort intellectuel de l'auteur, une condition qui est essentielle afin de donner à l'œuvre le caractère individuel requis. Selon l'article 2 de la [LPO], un programme d'ordinateur jouit de la protection s'il est original, en ce sens qu'il est une création intellectuelle propre à son auteur* »<sup>27</sup>. La cour estima qu'il n'existait aucune preuve d'infraction et ne se prononça pas davantage sur l'originalité du logiciel litigieux<sup>28</sup>.

Un peu plus tard, la même cour d'appel<sup>29</sup> dut se prononcer spécifiquement sur la question du critère d'originalité appliqué aux logiciels dans son arrêt du 19 décembre 2005. En l'espèce, les titulaires de droits d'auteur prétendaient que le critère d'originalité en matière de logiciel était moins strict qu'en matière de droit d'auteur traditionnel, en se basant sur le libellé de la LPO. La cour estima cependant que « *la question de savoir si l'exigence d'originalité découlant de la loi sur les logiciels va moins loin que l'exigence d'originalité imposée par le droit d'auteur n'est pas conforme à l'assimilation des programmes informatiques aux œuvres d'art ou aux œuvres littéraires* ». La cour poursuivit en précisant que le logiciel doit être original, aussi bien dans un sens objectif que subjectif, et doit dès lors bel et bien porter l'empreinte de la personnalité de l'auteur comme cela est exigé pour les œuvres

<sup>25</sup> Civ. Bruxelles (cess.), 30 juin 2003, *A&M*, 2004/2, p. 153 et s. On remarquera que cette considération répond à l'argument du demandeur, selon lequel « *un logiciel serait protégé dès qu'il est le résultat d'une création intellectuelle propre à l'auteur sans qu'il soit nécessaire que l'œuvre soit originale dans le sens de la loi « générale » sur le droit d'auteur du 30 juin 1994* »

<sup>26</sup> Cette définition avait déjà été retenue dans différentes décisions précédentes : voir Civ. Turnhout, 12 septembre 1994, *Computerr.*, 1995/2, p.65, note M. TAYMANS ; Corr. Hasselt, 16 février 1999, *I.R.D.I.*, 1999, p.34. Pour une décision « mitigée », voy. également Anvers (1<sup>ère</sup> ch.) 5 juin 2000, *A.J.T.*, 2000-2001, p. 919, note I. VERNIMME.

<sup>27</sup> Anvers (1<sup>ère</sup> ch.), 23 février 2004, *I.R.D.I.*, 2004, p.401 et s.

<sup>28</sup> Cependant, faisant référence à cette décision, un arrêt non publié de la cour d'appel de Gand aurait interprété ce passage en précisant qu'« *en vertu de l'article 2 de la [LPO], un programme d'ordinateur jouit de la protection s'il est original dans le sens où il est la création intellectuelle propre à l'auteur, sans qu'il soit exigé que l'œuvre soit originale au sens de la [LDA]* ». Gand (7<sup>ème</sup> ch.) 3 avril 2006 (Bvba Dekyver Sky Group - D. Dekyver - J. Dekyver / Nv Sky Service), R.G. 2004/AR/966, non publié et mentionné par J. Deene, « Het originaliteitscriterium in de Softwarewet », *Computerrecht*, 2007, p.148.

<sup>29</sup> Anvers (1<sup>ère</sup> ch.), 19 décembre 2005, *A&M*, 2007/1-2, p. 85 et s.

artistiques et littéraires traditionnelles. Ainsi, la cour reprend dans sa définition d'originalité applicable aux logiciels les éléments objectifs (effort intellectuel) et subjectifs (empreinte de la personnalité de l'auteur) de la notion traditionnelle. Dans l'application de ce principe au cas d'espèce, la cour semble cependant plus laxiste : elle retient que « *les paramètres des appelantes furent interprétés d'une façon bien déterminée, et il leur fut donné forme d'une façon déterminée* ». La cour constate également que le logiciel est le fruit « *de différents choix personnels que le programmeur dut faire* » et qu'il fit « *consciemment* »<sup>30</sup>. La cour en conclut que « *de ce qui précède, il découle que l'auteur a fait des choix personnels. Le critère subjectif est pareillement rempli. Le logiciel est l'expression de l'effort intellectuel de l'auteur* ». La cour semble dès lors estimer que le seul fait de prouver que des choix personnels ont été faits dans la programmation permet de remplir le critère subjectif d'originalité.

Qu'entend-t-on dès lors par le fait que l'effort intellectuel doit être « propre » à l'auteur du programme? Cette précision peut être pertinemment illustrée par un passage de l'arrêt de la Cour de cassation française du 7 mars 1986 : « *un effort personnalisé allant au-delà de la simple mise en œuvre d'une logique automatique et contraignante ... la matérialisation de cet effort résidant dans une structure individualisée* »<sup>31</sup>. Le travail intellectuel de l'auteur doit dès lors aller au-delà des contraintes et des lieux communs de la programmation, du simple « savoir-faire »<sup>32</sup>, et refléter les choix, l'organisation, la structure ou encore la technique propre de l'auteur. En d'autres termes, les lignes de codes imposées par la technique, ainsi que par toute autre contrainte externe, ne seront pas considérées comme étant originales, l'auteur n'ayant aucune possibilité de choix.

Le seuil d'originalité en matière de logiciel semble en fait facilement atteint, ce qui, au demeurant, correspond à l'interprétation qui est donnée au critère par la commission dans son rapport sur la mise en œuvre de la directive 91/250/CEE<sup>33</sup>. En effet, ce rapport constate que, pour la première fois, le seuil d'originalité a été harmonisé à l'échelle communautaire pour une catégorie spécifique d'œuvres, et que ce niveau uniforme a exigé de douze états membres, dont la Belgique<sup>34</sup>, un abaissement du seuil d'octroi de la protection.

Notons enfin que le 8<sup>ème</sup> considérant de la Directive précise que les critères appliqués pour déterminer si un programme d'ordinateur constitue ou non une œuvre originale ne devraient comprendre aucune évaluation de la qualité ou de la valeur esthétique du programme.

### Les logiciels FOSS ne sont protégés par le droit d'auteur que dans la mesure où ils sont originaux.

<sup>30</sup> « Bewust » dans le texte original.

<sup>31</sup> Cass. fr. (Ass. Plén.), 7 mars 1986, *JCP* 1986, II, 20631, note J.M. MOUSSERON, B. TEYSSIE et M. VIVANT; *RIDA* juillet 1986, n°129, p. 136, note A. LUCAS.

<sup>32</sup> A. LUCAS, *op. cit.*, p. 20 et s.

<sup>33</sup> Rapport de la Commission au Conseil, au Parlement européen et au Comité économique et social sur la mise en œuvre et les effets de la directive 91/250/CEE concernant la protection juridique des programmes d'ordinateurs, COM(2000), 199, p. 6.

<sup>34</sup> J. DEENE, *op.cit.*, p. 152.

De la même manière, un utilisateur n'aura besoin d'une licence que s'il reprend des éléments originaux d'un logiciel FOSS. Cela aura entre autres pour conséquence qu'une licence copyleft (cfr. *infra*) pourrait ne pas avoir d'effet dans la mesure où seule une partie non originale (et dès lors non-protégée par le droit d'auteur) pourrait avoir été reprise<sup>35</sup>.

Une des particularités des logiciels libres est le nombre de participants qui contribuent au développement d'un projet. Afin de pouvoir être considéré comme étant auteur du logiciel et bénéficiaire de droits d'auteur sur celui-ci, chacun des participants devra démontrer que son apport est original. Par exemple, un participant qui n'aurait que testé le logiciel en y répertoriant les bugs, ne bénéficiera pas de droits d'auteur sur le logiciel, peu importe l'assiduité de son travail. De même, il est peu probable que le participant qui n'aurait effectué qu'un travail de « débogage » voie son apport qualifié d'original<sup>36</sup>. Le doute s'installera cependant si à cette fin, le participant aura reformulé une partie du code.

## II.1.d. Les différents droits des auteurs

Les droits reconnus par la loi aux auteurs sont de deux types. On distingue en effet les droits patrimoniaux (davantage axés sur l'exploitation économique des œuvres) des droits moraux (censés préserver la personnalité de l'auteur et son lien avec l'œuvre). Ces deux types de droits naissent tous les deux dans le chef de l'auteur – personne physique – qui crée le logiciel, au moment où celui-ci est mis en forme.

Nous verrons cependant que certaines règles de présomption de transmission de droits aux employeurs existent (cfr. *infra*).

### II.1.d.1. Droits patrimoniaux

Les principaux droits patrimoniaux reconnus aux auteurs de logiciels sont les droits de reproduction, de distribution et de communication au public (voir l'article 5 de la LPO, couplé avec l'article 1§1 de la LDA)

---

<sup>35</sup> R.T. NIMMER, « Coexisting with Free and Open Source Software », *Computer und Recht International*, 2006, n°5, p. 131.

<sup>36</sup> Bien qu'il apparaisse, d'après une certaine jurisprudence française, que la qualité de coauteur puisse être reconnue à une personne en charge de la correction : voir A. LUCAS & H.-J. LUCAS, *Traité de la propriété littéraire et artistique*, Paris, Litec, 2<sup>ème</sup> édition, 2001, p. 66.

Il s'agit de droits « exclusifs » : seul le titulaire des droits d'auteur peut permettre ces activités, en soumettant éventuellement son autorisation à certaines conditions, ou en échange de certaines autres prestations. Ces droits sont dits « patrimoniaux » dans la mesure où ils ont trait à l'aspect commercial des œuvres et se négocient généralement en échange d'une rémunération (c'est entre autres le cas de la plupart des licences de logiciels « propriétaires »).

**Le droit de reproduction** consiste en la possibilité d'autoriser ou d'interdire toute reproduction (directe ou indirecte, en tout ou en partie) du logiciel sur tout support quelconque (disque dur, mémoire ROM, mémoire RAM, CD, DVD, clé USB, etc.). Il s'agit non seulement des reproductions permanentes ou durables (copie d'un logiciel gravé sur un CD) mais également des copies temporaires ou transitoires (chargement en mémoire vive). La notion de reproduction couvre aussi toute modification, adaptation, traduction, etc.

On notera que puisque l'utilisation de logiciels implique généralement systématiquement leur reproduction, à tout de moins partielle, en mémoire vive ou équivalent, on considère que le droit de reproduction permet, en fait, d'en contrôler l'usage. C'est la raison pour laquelle tout logiciel est systématiquement accompagné d'une licence autorisant la simple utilisation du logiciel. Le droit de reproduction est dès lors sans doute le droit ayant la plus grande portée en matière informatique<sup>37</sup>.

**Le droit de distribution** est le droit de distribuer au public tout support physique qui incorpore une reproduction (comme décrite précédemment), que ce soit par la vente, la location ou le prêt.

**La communication au public** est un concept « importé » du droit d'auteur traditionnel<sup>38</sup> (dans la mesure où il n'est pas repris explicitement dans la LPO) et implique toute transmission d'une œuvre autrement que par le biais de supports physiques. Elle vise entre autres la radiodiffusion, la télédiffusion/distribution, l'exécution d'une œuvre devant un public, etc. Des traités internationaux récents y ont également inclus « la mise à disposition du public de manière que chacun puisse y avoir accès de l'endroit et au moment qu'il choisit individuellement »<sup>39</sup>. Cette dernière disposition est particulièrement importante lorsqu'appliquée aux logiciels, car elle implique que toute mise à disposition ou diffusion du logiciel par le biais d'Internet est également couverte par le droit exclusif des auteurs.

---

<sup>37</sup> PH. LAURENT, « Les licences de type *open source* ou *open content* ne se fondent pas sur un « droit de destination » », *I.R.D.I.*, 2008, p.149 et s.

<sup>38</sup> A. STROWEL & E. DERCLAYE, *op. cit.*, p. 215, n° 243; F. DE VISSCHER & B. MICHAUX, *op. cit.*, p. 221, n° 259 ; F. BRISON & J.-P. TRIAILLE, « La nouvelle loi sur la protection des programmes d'ordinateurs, dans le sillage de la loi sur le droit d'auteur », *J.T.*, p. 143.

<sup>39</sup> Voir entre autre l'article 3 de la directive 2001/29/CE du Parlement européen et du Conseil du 22 mai 2001 sur l'harmonisation de certains aspects du droit d'auteur et des droits voisins dans la société de l'information.

Les licences FOSS sont des contrats de licences autorisant aux licenciés de reproduire, de modifier, de distribuer et de communiquer les logiciels ou leurs nouvelles versions. Cette autorisation est cependant soumise à certaines conditions (*cf. infra*).

On constatera que les licences FOSS sont souvent rédigées d'une façon assez générale, afin que l'autorisation donnée couvre tous les actes protégés par le droit d'auteur.

Cependant, certaines licences (la BSD ou la GPLv2 par exemple), basées sur le droit américain, ne portent que sur la « reproduction » et la « distribution », sans aborder la communication au public. Il faut cependant savoir qu'aux Etats-Unis, l'acte de distribution est conçu plus largement pour couvrir également la distribution par le biais de réseaux. En d'autres termes, la diffusion de logiciels par internet est incluse dans le concept de « distribution » américain. Il s'agirait dès lors – généralement, et afin de respecter la cohérence du système et la logique de l'open source - d'interpréter les termes de ces licences à la lumière du système législatif dont leur rédaction s'inspire, et dès lors, d'estimer que la diffusion des logiciels par le biais d'internet est également comprise dans le terme « distribution » tel qu'il y est employé.

On notera cependant que pareille application pourrait créer certaines frictions avec les règles d'interprétation spécifiques applicables en matière de cession de droits d'auteur (*cf. infra*).

## **II.1.d.2. Droits moraux**

Les droits moraux ont pour principale fonction de protéger la relation particulière et intime qui existe entre l'auteur et son œuvre. S'ils n'ont pas une finalité économique en soi, les droits moraux ont une importance particulière et peuvent également avoir des influences sur l'exploitation des œuvres.

En droit d'auteur classique, les droits moraux sont également au nombre de trois et rassemblent le droit d'attribution (ou « de paternité »), d'intégrité et de divulgation (voir article 1, §1 de la LDA).

La LPO ne prévoit qu'une référence au contenu minimal des droits moraux prévu à l'article 6bis de la Convention de Berne, qui reconnaît à l'auteur le droit de « revendiquer la paternité de l'œuvre » de même que le droit de « s'opposer à toute déformation, mutilation ou autre modification de cette œuvre ou à toute autre atteinte à la même œuvre, préjudiciable à son honneur ou à sa réputation ». Des doutes existent dès lors quant à l'étendue exacte des droits moraux sur les logiciels qui sont davantage expliqués ci-dessous.

Le **droit d'attribution (ou de paternité)** est en tout les cas reconnu aux auteurs des logiciels dans son acception traditionnelle. L'auteur a le droit d'exiger que son nom apparaisse sur l'œuvre (ou y soit à tout le moins associé), qu'il s'agisse de son vrai nom ou d'un pseudonyme. Il peut également exercer ce droit négativement et exiger de respecter son anonymat.

On remarquera que le droit de paternité est un droit moral d'une valeur particulièrement importante en matière de FOSS. L'une des clauses que l'on retrouve systématiquement dans les licences libres/open source est en effet celle qui oblige le respect des « copyright notices », qui comprennent généralement le nom de l'auteur et l'année de création.

Pareille reconnaissance (« *credit* ») est particulièrement importante dans une logique collaborative où la qualité et/ou l'importance des participations contribue au démarquage des profils et à la reconnaissance des expertises acquises.

Le **droit d'intégrité** prévu dans la LDA permet de s'opposer à toute modification ou déformation de l'œuvre, peu importe l'existence d'un préjudice dans le chef de l'auteur. Ce droit a été amoindri en matière de logiciels, car conformément à l'article 4 de la LPO, les auteurs de ceux-ci ne peuvent s'opposer aux modifications ou déformations que dans la mesure où celles-ci préjudicient son honneur ou sa réputation.

On notera que la question du droit moral d'intégrité sera particulièrement délicate en matière de licences FOSS. Nous avons vu que l'une des caractéristiques de ces licences est de permettre de modifier l'œuvre et de redistribuer la version modifiée. Normalement, sauf mention explicite dans la licence, cette autorisation doit être considérée comme étant donnée sur la seule base du droit patrimonial de reproduction, les droits moraux étant conservés par l'auteur<sup>40</sup>.

Les licenciés ne pourraient dès lors pas apporter n'importe quelle modification au logiciel... l'auteur ayant toujours le droit de s'opposer aux modifications qui porteraient atteinte à son honneur ou sa réputation. Cependant, le caractère éminemment fonctionnel des programmes d'ordinateurs rend mal aisée l'appréciation d'une atteinte à l'honneur ou à la réputation de l'auteur.

Il n'est pas certain que les auteurs de logiciels bénéficient du droit moral de **divulgation**, qui est la prérogative de l'auteur de décider quand son œuvre est achevée et peut être divulguée au public. En effet, la LPO ne prévoyant de droits moraux que par seule référence à l'article 6bis de la Convention de Berne (qui ne

---

<sup>40</sup> F. DE PATOUL, « Logiciels libres et droit d'auteur : les droits moraux et les règles contractuelles », in *Les logiciels libres face au droit*, Bruxelles, Bruylant, 2005, p. 108.

prévoit pas ce droit), la doctrine est divisée sur la question de savoir si seuls les droits moraux prévus par cette convention doivent être retenus, ou si, au contraire, il faut estimer que, puisque le droit de divulgation n'est pas abordé par la LPO, il s'agit de se référer à la LDA qui, elle, le prévoit<sup>41</sup>. Dans le doute, le praticien partira du principe que ce droit doit également être reconnu aux auteurs de logiciels.

On notera qu'en pratique, lorsque l'on analyse la jurisprudence, on peut constater que les droits moraux des auteurs de logiciels, visés à l'article 4 de la LPO, ne semblent pas souvent être invoqués, ou du moins, faire l'objet de débats passionnés. La violation des droits moraux n'est généralement invoquée qu'en surplus de toutes les violations d'ordre patrimonial, et semble tout au plus faire l'objet d'une certaine reconnaissance par les juges. Par exemple, le Tribunal de première instance de Gand, dans un cas de contrefaçon manifeste (*cf. infra*), remarqua que des dommages moraux avaient été subis « *entre autres, par l'atteinte au produit et au nom de la demanderesse* »<sup>42</sup> sans pour autant en faire découler des conséquences spécifiques dans la condamnation du contrefacteur.

Nous avons cependant remarqué que, comme les droits patrimoniaux faisaient l'objet d'une permissivité tout à fait remarquable dans le cadre des licences FOSS, l'usage du droit moral de paternité se démarque, et que, de même, le droit moral d'intégrité prend davantage d'importance.

### II.1.d.3. Exceptions aux droits d'auteur

Traditionnellement, tous les régimes de droits exclusifs constituant la « propriété intellectuelle » sont assortis d'exceptions à ces droits, par considération pour des droits, libertés et/ou intérêts conflictuels (tels que le droit de la vie privée, le droit à l'information, l'éducation, la promotion culturelle et sociale,...)

La LDA prévoit une série d'exceptions aux droits d'auteur, dans ses articles 21 et 22.

De même, les articles 6 et 7 de la LPO prévoient des exceptions spécifiques aux droits d'auteur sur les programmes d'ordinateur.

L'article 6 §1 de la LPO<sup>43</sup> prévoit une **exception d'utilisation**. Celle-ci permet à un utilisateur légitime<sup>44</sup> d'utiliser le programme d'ordinateur conformément à sa destination<sup>45</sup>.

<sup>41</sup> Voy. à propos de cette controverse, F. DE PATOUL, *op. cit.*, p. 96 et s.

<sup>42</sup> Civ. Gand (1<sup>ère</sup> ch.), 23 mai 2005, *I.R.D.I.*, 2005, p. 516 et s.

<sup>43</sup> « En l'absence de dispositions contractuelles particulières, ne sont pas soumis à l'autorisation du titulaire les actes visés à l'article 5, a) et b), lorsque ces actes sont nécessaires pour permettre à la personne ayant le droit d'utiliser le programme d'ordinateur, de l'utiliser d'une manière conforme à sa destination, en ce compris la correction d'erreurs ».

L'article 6 §2 de la LPO<sup>46</sup> prévoit une exception de « backup » ou de **copie de sauvegarde**.

L'article 6 §3 de la LPO<sup>47</sup> prévoit une **exception d'analyse** des fonctionnalités du programme.

L'article 7 de la LPO<sup>48</sup> prévoit une **exception de décompilation** particulièrement complexe.

La doctrine se divise sur la question de l'exhaustivité des exceptions reprises dans la LPO. En effet, alors que certains auteurs<sup>49</sup> estiment que seules les exceptions spécifiques de la directive (reprises dans la LPO) sont applicables en matière de programmes d'ordinateur, d'autres estiment que certaines des exceptions prévues aux articles 21 et suivants de la LDA doivent également s'appliquer. A la lumière du vingt-huitième considérant de la Directive, une doctrine majoritaire en Belgique

---

<sup>44</sup> La cour d'appel d'Anvers a eu l'occasion de se prononcer brièvement sur la notion d' « utilisateur légitime », qui fut invoquée par une personne accusée de violation des droits sur le logiciel litigieux. Selon la cour, la LPO et la directive 91/250/CEE prévoient seulement que peut être considéré comme étant un « utilisateur légitime » celui qui a reçu l'autorisation d'utiliser le logiciel d'un acheteur des droits ou d'un licencié. La cour conclut que « la notion « d'utilisateur légitime » ne peut pas être étendue à quiconque qui aurait reçu une autorisation d'un prétendu utilisateur légitime au second degré », Anvers (1<sup>ère</sup> ch.), 19 décembre 2005, *A&M*, 2007/1-2, p. 85 et s.

<sup>45</sup> La notion exacte de la « destination d'un logiciel » est controversée. Voir PH. LAURENT, « Logiciels libres et droit d'auteur : naissance, titularité et exercice des droits patrimoniaux », in *Les logiciels libres face au droit*, Bruxelles, Bruylant, 2005, p. 78 et s.

<sup>46</sup> « La personne ayant le droit d'utiliser le programme d'ordinateur ne peut s'en voir interdire la reproduction sous la forme d'une copie de sauvegarde pour autant que cette copie soit nécessaire à l'utilisation du programme ».

<sup>47</sup> « La personne ayant le droit d'utiliser le programme d'ordinateur peut, sans l'autorisation du titulaire du droit, observer, étudier ou tester le fonctionnement de ce programme afin de déterminer les idées et les principes qui sont à la base d'un élément du programme, lorsqu'elle effectue une opération de chargement, d'affichage, de passage, de transmission ou de stockage du programme d'ordinateur qu'elle est en droit d'effectuer ».

<sup>48</sup> « § 1. L'autorisation du titulaire du droit n'est pas requise lorsque la reproduction du code ou la traduction de la forme de ce code au sens de l'article 5, a) et b) est indispensable pour obtenir les informations nécessaires à l'interopérabilité d'un programme d'ordinateur créé de façon indépendante avec d'autres programmes et sous réserve que les conditions suivantes soient réunies :

- a) les actes de reproduction et de traduction sont accomplis par une personne jouissant du droit d'utiliser une copie du programme, ou, pour son compte, par une personne habilitée à cette fin;
- b) les informations nécessaires à l'interopérabilité ne lui sont pas déjà facilement et rapidement accessibles;
- c) les actes de reproduction et de traduction sont limitées aux parties du programme d'origine nécessaires à cette interopérabilité.

§ 2. Les dispositions du paragraphe précédent ne peuvent justifier que les informations obtenues en vertu de leur application :

- a) soient utilisées à d'autres fins que la réalisation de l'interopérabilité du programme créé de façon indépendante;
- b) soient communiquées à des tiers, sauf si ces communications s'avèrent nécessaires à l'interopérabilité du programme d'ordinateur créé de façon indépendante;
- c) ou soient utilisées pour la mise au point, la production ou la commercialisation d'un programme d'ordinateur dont l'expression est fondamentalement similaire, ou pour tout autre acte portant atteinte au droit d'auteur.

§ 3. Le présent article ne peut recevoir une application qui cause un préjudice injustifié aux intérêts légitimes du titulaire du droit, ou porte atteinte à l'exploitation normale du programme d'ordinateur ».

<sup>49</sup> B. CZARNOTA & R.J. HART, *Legal Protection of Computer Programs in Europe*, Londres, Butterworth, 1991, p. 31.

considère que les exceptions de la LDA s'appliquent également en matière informatique, sauf lorsqu'une exception est déjà abordée par la LPO (à savoir, principalement, la copie privée, remplacée dans la LPO par la possibilité de faire une copie de sauvegarde)<sup>50</sup>. Les exceptions de la LDA qui présentent un certain intérêt en ce qui concerne les logiciels seront entre autres la citation dans un but de critique, de polémique ou d'enseignement et la reproduction de courts fragments à l'occasion de comptes rendus des événements d'actualité.

L'intérêt des exceptions aux droits d'auteur est très limité lorsque ces droits font l'objet d'une licence FOSS. En effet, les licences FOSS étant très permissives, et les code sources des logiciels étant fournis, les actes autorisés vont généralement bien au-delà de ce que les exceptions prévoient.

Cependant, certaines situations pourraient se présenter où, pour une raison ou une autre, des utilisateurs de logiciels FOSS préféreraient invoquer une exception plutôt que d'accepter les termes de la licence.

## II.1.e. Titularité des droits d'auteur

La détermination des titulaires des droits d'auteur est essentielle dans la mesure où ce sont ces derniers qui bénéficient des droits, prérogatives et protections accordés par le droit d'auteur, ainsi que des actions en justice qui permettront le respect de ces droits.

En pratique, s'agissant de logiciels FOSS, ce seront les titulaires des droits qui, entre autres, choisiront la licence sous laquelle seront distribués les logiciels et qui auront l'occasion de poursuivre toute personne qui ne respecte pas les obligations de cette licence.

### II.1.e.1. Auteur originaire

La règle générale en matière de titularité des droits d'auteur est que ceux-ci reviennent à **la personne physique** qui a créé l'œuvre (art. 6 §1 de la LDA).

Lorsqu'il est fait référence au créateur de l'œuvre, il s'agit avant tout de la personne qui l'a intellectuellement conçue dans tout ce qu'elle a d'original. C'est en effet l'apport d'élément(s) original(aux) qui déterminera l'auteur et non le simple fait de

<sup>50</sup> F. DE VISSCHER & B. MICHAUX, *op. cit.*, p. 223 et s. ; A. STROWEL & E. DERCLAYE, *op. cit.*, p. 243 ; F. BRISON & J.-P. TRIAILLE, *op. cit.*, p. 143.

« travailler sur l'œuvre ». Ainsi, par exemple, c'est normalement l'architecte qui bénéficie de la protection par les droits d'auteur sur un édifice original issu de son imagination, et non les ouvriers qui le construisent.

Il est fait référence à une personne physique (c'est-à-dire de chair et de sang) afin de l'opposer aux personnes morales, telles que les sociétés. En effet, l'auteur originaire ne peut jamais, en Belgique, être une société : les droits naîtront toujours dans le chef des hommes et des femmes qui y travaillent. Nous verrons cependant qu'il existe, dans certains cas, une présomption de cession (cfr. infra).

## II.1.e.2. Collaboration de plusieurs auteurs

Plusieurs auteurs peuvent contribuer à la création d'une œuvre commune, en y apportant chacun des éléments originaux. Il s'agit dans ce cas d'une **œuvre de collaboration**, c'est-à-dire d'une « œuvre à la création de laquelle ont concouru plusieurs personnes physiques »<sup>51</sup>. Il n'y a œuvre de collaboration que lorsqu'il y a « unicité de l'œuvre », c'est-à-dire, que les auteurs ont véritablement joint leurs efforts pour créer, volontairement ensemble, une œuvre commune. Ceci implique une concertation certaine entre les auteurs<sup>52</sup>. Ce sera le cas, par exemple, lorsque plusieurs développeurs décideront ensemble de développer un programme commun et de se répartir les tâches.

La LDA prévoit certaines règles en matière d'œuvre de collaboration<sup>53</sup>. Nous retiendrons que le statut de l'œuvre est celui d'une indivision, et que les auteurs doivent dès lors s'accorder, entre autres, sur la manière de l'exploiter. De même, ils exerceront ensemble les droits moraux relatifs à l'œuvre, et, dès lors, décideront du moment où l'œuvre peut être divulguée, et sous quels noms.

---

<sup>51</sup> En l'absence de définition dans la loi belge, il est fréquemment fait référence à cette définition extraite de l'article L. 113-2 du Code français de la Propriété Intellectuelle.

<sup>52</sup> Voir PH. LAURENT, *op. cit.*, p. 32.

<sup>53</sup> L'article 4 de la LDA prévoit : « Lorsque le droit d'auteur est indivis, l'exercice de ce droit est réglé par les conventions. A défaut de conventions, aucun des auteurs ne peut l'exercer isolément, sauf aux tribunaux à se prononcer en cas de désaccord. Toutefois, chacun des auteurs reste libre de poursuivre, en son nom et sans l'intervention des autres, l'atteinte qui serait portée au droit d'auteur et de réclamer des dommages et intérêts pour sa part.

*Les tribunaux pourront toujours subordonner l'autorisation de publier l'œuvre aux mesures qu'ils jugeront utiles de prescrire; ils pourront décider à la demande de l'auteur opposant, que celui-ci ne participera ni aux frais, ni aux bénéfices de l'exploitation ou que son nom ne figurera pas sur l'œuvre. »*

L'article 5 de la LDA prévoit : « Lorsqu'il s'agit d'une œuvre de collaboration où la contribution des auteurs peut être individualisée, ces auteurs ne peuvent, sauf convention contraire, traiter de leurs œuvres avec des collaborateurs nouveaux.

*Néanmoins, ils auront le droit d'exploiter isolément leur contribution, pour autant que cette exploitation ne porte pas préjudice à l'œuvre commune. »*

En matière de logiciels libres, les cas d'œuvres de collaborations seront assez fréquents<sup>54</sup>. Beaucoup de projets de développement sont lancés par plusieurs développeurs qui s'accordent sur leurs objectifs et se répartissent le travail. Pareils auteurs qui développent communément un logiciel auront, entre autres, la tâche importante de décider ensemble de la licence sous laquelle le logiciel sera distribué.

### II.1.e.3. Œuvres dérivées

Lorsqu'une œuvre est créée et divulguée au public par son (ses) auteur(s), nous avons vu que ce(s) dernier(s) a (ont) le droit d'en faire respecter l'intégrité et de s'opposer à toute modification de celle-ci.

Cependant, le (les) titulaire(s) de droits peuvent également autoriser à des tiers de créer des œuvres dérivées. L'**œuvre dérivée** est une « œuvre nouvelle à laquelle est incorporée une œuvre préexistante sans la collaboration de l'auteur de cette dernière »<sup>55</sup>.

Les différences entre l'œuvre de collaboration et l'œuvre dérivée sont fondamentales : alors que l'œuvre de collaboration est une œuvre commune créée entre plusieurs auteurs en concertation, l'œuvre dérivée est une nouvelle œuvre, distincte de l'œuvre originale (il n'y a donc pas d'unicité avec l'œuvre originale), et créée par un (ou plusieurs) autre(s) auteur(s) sans concertation avec l' (les) auteur(s) de l'œuvre originale (mais bien avec son (leur) autorisation).

Cette différence a également d'importantes conséquences au niveau de la titularité des droits d'auteurs sur l'œuvre dérivée et sur son exploitation. L'auteur de l'œuvre dérivée est le seul auteur de cette nouvelle œuvre. En d'autres termes, l'auteur de l'œuvre originale ne doit pas être considéré comme étant co-auteur de l'œuvre dérivée. Par ailleurs, seul l'auteur de l'œuvre dérivée décide de la façon de l'exploiter (mais il devra veiller à respecter les obligations auxquelles il se serait engagé dans le cadre de l'autorisation reçue de l'auteur de l'œuvre originale, ainsi que les limites de cette licence).

Autrement dit, l'œuvre dérivée doit être considérée comme une œuvre distincte à part entière, la seule différence étant que son auteur doit respecter le contrat (de licence) passé avec l'auteur de l'œuvre originale.

---

<sup>54</sup> Y. VAN DEN BRANDE & J. KEUSTERMANS, "Open source software: een analyse naar Belgisch recht", *I.R.D.I.*, 2007, p.374.

<sup>55</sup> En l'absence de définition dans la loi belge, il est fréquemment fait référence à cette définition extraite de l'article L. 113-2 du Code français de la Propriété Intellectuelle.

Les licences FLOSS sont de véritables licences de création « en série » d'œuvres dérivées<sup>56</sup>, l'autorisation de modifier le logiciel et d'en redistribuer les modifications étant la base de tout le système.

L'évolution d'un projet open source n'est en fait qu'une succession d'œuvres dérivées, qui peuvent également, à chaque étape, consister en des œuvres de collaboration. La première version du logiciel peut en effet être créée par une personne seule, ou par plusieurs développeurs (œuvre de collaboration). Elle sera publiée sous une licence libre permettant sa modification et la redistribution de la nouvelle version. Grâce à cette licence un développeur, ou un autre groupe de développeurs (œuvre de collaboration) - qui peut, par ailleurs, comprendre bon nombre des développeurs originaires - se servira de cette première version pour créer une autre version, dérivée de la première.... et ainsi de suite.

La notion d' « œuvre dérivée », ou celle de logiciel développé sur la base d'un logiciel précédent (« work based on the program ») sont des notions essentielles en matière de FOSS. En effet, c'est sur ces concepts que reposent d'importantes obligations, et entre autres, le déclenchement des clauses copyleft (cfr. infra).

En raison de la nature même de la programmation informatique<sup>57</sup>, ces concepts sont cependant assez flous et soumis à l'appréciation des juges, sans qu'il n'existe pour l'instant de jurisprudence éclairante à ce sujet. Par ailleurs, les avis divergent sur l'application de ces règles à différentes techniques spécifiques, dont, entre autres, les bibliothèques, les drivers et leurs relations avec d'autres logiciels y faisant appel.

Ces notions d'œuvre dérivée et d'œuvre « basée sur le programme » devraient faire l'objet d'une analyse spécifique détaillée.

#### **II.1.e.4. Transfert, présomptions de transfert et conséquences**

Si les droits d'auteur naissent dans le chef de leur créateur, personne physique, ils peuvent cependant faire l'objet d'un transfert (du moins, les droits patrimoniaux<sup>58</sup>). Il est donc fréquent que les auteurs cèdent l'entièreté de leurs droits patrimoniaux à

---

<sup>56</sup> PH. LAURENT, *op.cit.*, p.52 et s.

<sup>57</sup> B. W. CARVER, « Share & Share Alike : Understanding and Enforcing Open Source and Free Software Licenses », *Berkeley Technology Law Journal*, Vol. 20:443, p.458.

<sup>58</sup> La question est plus délicate en matière de droits moraux. La LDA prévoit en son article 1 que le droit moral est « inaliénable » et que « la renonciation globale à l'exercice de ce droit est nulle ». L'application de ce principe aux droits d'auteur relatifs aux programmes d'ordinateur est cependant débattue en doctrine. Voir par exemple à ce sujet A. STROWEL & E. DERCLAYE, *op. cit.*, p. 212.

d'autres personnes ou à des sociétés. Dans ce cas, il appartiendra à ces cessionnaire de décider de l'exploitation de ces droits.

La question de la titularité des droits d'auteur sur un logiciel n'est traitée, dans la LPO, qu'à l'article 3. Ce dernier ne porte que sur une situation précise, à savoir les **créations d'employés**. Contrairement à la LDA, la LPO prévoit en effet une présomption réfragable (applicable sauf preuve contraire, ou sauf mention contraire dans le contrat de travail) de cession à l'employeur des droits patrimoniaux relatifs aux programmes d'ordinateur créés par les employés dans l'exercice de leurs fonctions ou d'après les instructions de leur employeur<sup>59</sup>.

Dans ce cas, les droits naissent dans le chef de l'employé, mais sont considérés comme étant immédiatement transférés à son employeur.

Cette règle a pour importante conséquence que lorsqu'un développement est effectué en entreprise, c'est cette dernière qui est généralement (sauf clause contraire dans le contrat de travail) titulaire des droits d'auteur, et qui peut, dès lors, décider de l'exploitation du logiciel et de sa licence.

Dans ce cas, il n'appartient donc pas à l'employé de décider ou non de distribuer le logiciel sous licence FLOSS. A supposer que l'employé le fasse quand même, pareille licence serait nulle.

La LPO ne prévoit pas une présomption similaire en cas de **contrat de commande** : c'est dès lors l'article 3, §3, al. 2 de la LDA (concernant les règles contractuelles générales) qui est d'application dans ce cas<sup>60</sup> (cfr. infra).

<sup>59</sup> On comprendra pourquoi l'employeur d'un créateur d'un site web essaya, en vain, de faire passer ce site web pour un logiciel afin de tenter de bénéficier de pareille présomption. Voir Civ. Gand (cess.), 3 septembre 2001, *I.R.D.I.*, 2002, p. 104 et s., note E. LAEVENS

<sup>60</sup> Dans un cas d'espèce, le président du tribunal de première instance de Bruxelles souligna que cet article porte sur les contrats de commande liant un auteur à son commanditaire. Se référant à l'article 6 de la LDA, le président constata que la société titulaire des droits d'auteur n'était pas une personne physique, et ne pouvait dès lors n'être qu'un cessionnaire de ceux-ci (soit par l'effet d'un contrat de cession passé avec l'auteur originaire, soit en application de l'article 3 de la LPO). Cette société n'ayant pas la « qualité d'auteur » (au sens strict), le président décida de se référer au droit commun des contrats, et souligna que « la cession des droits d'utilisation portant sur les logiciels litigieux peut être tacite, pourvu qu'elle soit certaine », ce qu'il retint dans le cas d'espèce. Voir Civ. Bruxelles (cess.), 25 janvier 2002, *A&M*, 2004/4, p. 333 et s.

Cette différence de traitement entre les logiciels développés par des employés et les logiciels développés sous contrat de commande a également été soulignée par le juge des saisies de Bruxelles du 17 octobre 2003. Le cas qui lui fut soumis concernait une opposition sur l'ordonnance de saisie description d'un logiciel opérée par son prétendu auteur. Ce dernier était titulaire d'une licence sur le logiciel en question, et avait également demandé au donneur de licence d'apporter des modifications au logiciel. Il prétendit dès lors être titulaire des droits sur ces modifications en application de l'article 3 de la LPO. Le juge des saisies accepta le principe selon lequel l'adaptation d'un logiciel peut effectivement être protégée par le droit d'auteur si cette adaptation, en tant que telle, est originale, et, dans ce cas, il s'en suit que ce droit d'auteur naît dans le chef de la personne qui l'a développée. Le juge des saisies rappela ensuite que l'article 3 de la LPO ne prévoit une présomption de cession qu'en matière de créations par des employés et non en matière de contrats de commande, comme c'était le cas en l'espèce. Appliquant dès lors l'article 3, §3, al. 2 de la LDA, il constata que le saisissant n'apportait aucune

## II.1.e.5. Présomption de titularité

L'article 6, al.2 de la LDA prévoit qu' « *est présumé auteur, sauf preuve contraire, quiconque apparaît comme tel sur l'œuvre, sur une reproduction de l'œuvre ou en relation avec une communication au public de celle-ci, du fait de la mention de son nom ou d'un sigle permettant de l'identifier* ».

Cette présomption permet entre autres aux tiers de partir du principe que les noms mentionnés dans les notices de copyrights sont bien les noms des titulaires de droits, et donc, des personnes à contacter pour obtenir une licence sur l'œuvre. Cette présomption n'a cependant pas d'effet dans le cadre d'un litige concernant la titularité d'un logiciel en tant que telle : lorsque deux personnes prétendront être auteurs d'une œuvre, la mention du nom de l'un d'entre eux n'enclenchera pas la présomption vis-à-vis de l'autre.

En ce qui concerne l'applicabilité de cette règle aux logiciels, on notera que la cour d'appel d'Anvers a confirmé que « *la présomption de l'article 6, al. 2, de la LDA vaut également pour un programme informatique bénéficiant de la protection du droit d'auteur si le nom de la firme apparaît sur l'écran lors de l'utilisation du logiciel* »<sup>61</sup>.

## II.1.f. Durée des droits d'auteur

Les droits d'auteurs s'éteignent 70 ans après la mort de l'auteur originaire d'une œuvre (art.2, §1 de la LDA).

En cas d'œuvre de collaboration, cette période s'étend à 70 ans après la mort du dernier des co-auteurs (art.2, §2 de la LDA).

Les mêmes durées sont applicables aux logiciels.

---

preuve de cession des droits, et donna raison à l'opposant. Voy. Juge des saisies de Bruxelles, 17 octobre 2003, *I.R.D.I.*, 2004, p. 160 et s.

On notera qu'une certaine doctrine regrette qu'une cession automatique des droits au client ne soit pas prévue ou facilitée par la loi (faute d'harmonisation par la directive) en cas de contrat de commande. Voy. J.-P. TRIAILLE, « L'application de la loi aux logiciels », *A&M*, 2004/5-6, p. 435.

<sup>61</sup> Anvers (1<sup>ère</sup> ch.), 19 décembre 2005, *A&M*, 2007/1-2, p. 85 et s.

## II.2. Droit des brevets

Malgré le fait que le droit d'auteur soit la protection « naturelle » des logiciels, ceux-ci font de plus en plus l'objet de demandes de brevet<sup>62</sup>.

Dès lors, le droit d'auteur n'est pas la seule branche juridique impliquée dans la rédaction ou l'étude des effets des licences FOSS, le droit des brevets faisant de plus en plus l'objet de réactions vives au sein des communautés, et ayant pour conséquence l'ajout de clauses défensives (parfois même offensives) dans les contrats de licence.

### II.2.a. Plusieurs systèmes et bases légales multiples

Le droit des brevets est une matière complexe et réglementée à différents niveaux. Le brevet est un titre qui est décerné par une institution spécifique pour un territoire déterminé. Il est possible d'obtenir des brevets par différents créneaux.

Les brevets sont avant tout des titres décernés par chaque pays pour la couverture de leurs territoires. En Belgique, c'est l'office de la propriété intellectuelle (l'[OPRI](#)) qui délivre les **brevets belges**, en suivant les règles établies dans le cadre de la loi du 23 mars 1984 sur le brevet d'invention (la [LBI](#)).

L'organisation européenne des brevets, instaurée par la Convention Européenne des Brevets (la [CEB](#)), regroupe actuellement 36 pays d'Europe, et a mis en place un système de **brevets européens**. C'est l'office européen des brevets (l'[OEB](#)) qui a pour tâche de décerner et administrer ces brevets.

Enfin, il existe une **procédure internationale** de dépôt de brevet, prévue dans le cadre du « Patent Cooperation Treaty » (le [PCT](#)), organisé et implémenté par l'office mondial de la propriété intellectuelle (l'[OMPI](#)).

Etant donné la multiplicité des systèmes et la complexité de la matière, la présente étude n'aura pour seul objectif que d'en exposer les grands principes communs en Europe.

---

<sup>62</sup> Voir par exemple B. DE VUYST & A. FAIRCHILD, « Legal and Economic Justification for Software Protection » in *Software Applications, Concepts, Methodologies, Tools and Applications*, Vol.4, Information Science Reference, New York, 2009, p.2831 et s.

Le "brevet d'invention", est un droit exclusif et temporaire d'exploitation décerné pour une invention qui est nouvelle, implique une activité inventive et est susceptible d'application industrielle<sup>63</sup>. Ci-dessous sont développées les notions d'invention, les conditions de brevetabilité et l'étendue des droits exclusifs.

## II.2.b. Objet de la protection : l'invention

### II.2.b.1. Notion d'invention

Les brevets ne peuvent être obtenus que sur des inventions, à savoir des solutions techniques à des problèmes techniques. Ils peuvent normalement être obtenus dans n'importe quel domaine technologique<sup>64</sup>.

Le caractère technique est en effet ce qui distingue l'objet protégé par les brevets des objets protégés par d'autres droits intellectuels (œuvres littéraires et artistiques, marques, dessins et modèles, etc.) ou les objets dépourvus de protection (les découvertes, les formules mathématiques, etc...)

Les inventions sont divisées en deux catégories : les inventions de produits et les inventions de procédés.

L'invention de produit porte sur « un corps déterminé ayant une composition mécanique ou une structure chimique particulière »<sup>65</sup>.

L'invention de procédé porte sur un moyen incorporel, une manière de faire, une formule, une succession d'opération (voire une opération), qui mènent à un résultat ou un produit, connu ou nouveau<sup>66</sup>.

### II.2.b.2. Brevetabilité des logiciels ?

Les textes législatifs prévoient traditionnellement des listes d'éléments qui ne sont pas des inventions, et qui ne peuvent dès lors pas faire l'objet de brevets.

---

<sup>63</sup> Art. 2 de la LBI.

<sup>64</sup> Art. 52 § 1 de la CEB.

<sup>65</sup> M. BUYDENS, *Droit des brevets d'invention*, Bruxelles, Larcier, 1999, p. 53, citant ROUBIER, *Le droit de la propriété intellectuelle*, Paris, 1954, t.2, n°142 et VAN REEPINGHEN & DE BRABANTER, *Les brevets d'invention*, Bruxelles, 1987, n°8.

<sup>66</sup> M. BUYDENS, *op. cit.*, p. 53.

Ainsi, l'article 52, §§ 1 et 2 de la CEB prévoit :

« §2. Ne sont pas considérés comme des inventions au sens du paragraphe 1 notamment :

- a) les découvertes, les théories scientifiques et les méthodes mathématiques ;
- b) les créations esthétiques ;
- c) les plans, principes et méthodes dans l'exercice d'activités intellectuelles, en matière de jeu ou dans le domaine des activités économiques, ainsi que les programmes d'ordinateur ;
- d) les présentations d'informations.

§3 Le paragraphe 2 n'exclut la brevetabilité des éléments qu'il énumère que dans la mesure où la demande de brevet européen ou le brevet européen concerne l'un de ces éléments, considéré en tant que tel. »

Sur la base de cette exclusion de la brevetabilité des « programmes d'ordinateur en tant que tels », l'OEB a développé une jurisprudence établissant certains critères visant à distinguer les logiciels exclus du champ de brevetabilité de ceux qui ne le sont pas (car considérés comme sortant de la catégorie « logiciel en tant que tel »).

Cette pratique est décrite par l'OEB en ces termes (traduits de l'anglais<sup>67</sup>):

« L'OEB ne concède pas de brevet pour les programmes d'ordinateurs ou les méthodes commerciales qui n'apportent aucune contribution technique. Les programmes d'ordinateurs en tant que tels sont exclus du champ de brevetabilité en application des art. 52(2)(c) et (3) de la CEB. Conformément à cette loi, un programme d'ordinateur n'est pas brevetable s'il n'a pas la capacité de causer un "effet technique supplémentaire" qui doit aller au delà des interactions techniques inhérentes entre logiciel et hardware.

D'un autre côté, une invention mise en œuvre par ordinateur (même sous la forme d'un programme d'ordinateur) qui peut fournir cet effet technique

<sup>67</sup> "The EPO does not grant patents for computer programs or computer-implemented business methods that make no technical contribution. Programs for computers as such are excluded from patentability by virtue of Art. 52(2)(c) and (3) EPC. According to this patent law, a program for a computer is not patentable if it does not have the potential to cause a "further technical effect" which must go beyond the inherent technical interactions between hardware and software.

On the other hand, a CII [lire "Computer Implemented Invention] (even in the form of a computer program) that can provide this further technical effect can be patentable, subject to the other patentability requirements, such as novelty and inventive step. In this case, it would be recognised as providing a technical solution to a technical problem." Citation reprise d'une page web de l'OEB à l'adresse [http://www.epo.org/topics/issues/computer-implemented-inventions/software\\_fr.html](http://www.epo.org/topics/issues/computer-implemented-inventions/software_fr.html). Voir également le document de l'OEB « Des brevets pour les logiciels », OEB, 2009, disponible on-line.

*supplémentaire peut être brevetable, à condition qu'elle remplisse les autres conditions de brevetabilité, telles que la nouveauté et l'activité inventive. Dans ce cas, elle sera reconnue comme constituant une solution technique à un problème technique».*

Cette pratique de l'OEB implique dès lors que, dans des cas déterminés et à certaines conditions, les logiciels peuvent faire l'objet d'un brevet.

Un logiciel peut tout d'abord faire l'objet d'un brevet, lorsque ce dernier porte sur un objet incorporant un logiciel, ou sur un procédé technique dont l'exécution se fait par le biais d'un logiciel: le logiciel ne fait alors que partie de l'invention visée. L'exemple classique de pareille situation est le système de freinage de type « ABS » installé dans les voitures et contrôlé électroniquement grâce à un logiciel « embarqué ».

Mais l'OEB va plus loin et accepte de breveter des logiciels, même installés sur des ordinateurs comme des programmes « classiques », à condition que leur effet soit un effet technique allant au-delà des effets dus à l'usage d'un logiciel sur l'ordinateur. C'est par exemple le cas, suivant la jurisprudence de l'OEB, lorsqu'un logiciel permet d'améliorer le résultat final d'images numériques, d'effectuer un traitement d'entités physiques, ou encore d'assurer une transmission de données d'un programme à un autre.

### **II.2.b.3. Portée de la protection en ce qui concerne les logiciels**

A l'inverse du droit d'auteur, le brevet protège davantage le fond que la forme.

Alors que le droit d'auteur protège une forme d'expression spécifique d'une idée (qui, elle, reste dans le domaine public car non protégée en tant que telle), les brevets protègent des « inventions », à savoir des solutions techniques à des problèmes techniques.

L'invention implémentée par ordinateur sera généralement une invention de procédé<sup>68</sup>. Nous avons vu qu'il s'agit dès lors de protéger une « manière de faire » ou une ou plusieurs « opérations ». Dans le domaine du logiciel, cela se traduira par une protection accordée aux fonctionnalités, peu importe la façon dont celles-ci sont codées.

Dès lors, alors que le droit d'auteur ne protège que le code, le brevet protégera la fonctionnalité que le code implémente (peu importe sa « forme » et donc peu importe la façon dont ce code est rédigé ou construit). Il en résulte qu'alors qu'une violation de droit d'auteur peut être évitée en créant une nouvelle forme, la violation d'un brevet ne peut s'éviter qu'en s'abstenant d'utiliser la fonction qui fait l'objet du brevet.

<sup>68</sup> S. DUSOLLIER, “Le brevet de logiciel qui ne dit pas son nom”, *I.R.D.I.*, 2007, p.246.

Par ailleurs, on notera qu'alors qu'il est normalement impossible de violer les droits d'auteur d'une œuvre dont on n'a pas connaissance (la protection portant sur l'originalité, qui dépend de la personnalité – normalement unique – de chaque individu), un développeur court davantage de risque de rencontrer un problème technique, et d'aboutir à la même solution technique que celle protégée par un brevet dont il n'aurait pas connaissance.

C'est la raison pour laquelle les communautés du « libre » sont généralement largement opposées à la brevetabilité des logiciels, qui crée, dans le chef des développeurs, certains risques de violer sans le savoir la propriété intellectuelle d'autrui, et qui limite les possibilités de développement de solutions/fonctionnalités informatiques.

## II.2.c. Conditions de brevetabilité

Un brevet n'est accordé à une invention que si elle est nouvelle, qu'elle le fruit d'une activité inventive et qu'elle peut faire l'objet d'une application industrielle.

Une invention est considérée comme **nouvelle** si elle n'est pas comprise dans l'état de la technique. L'état de la technique est constitué par tout ce qui a été rendu accessible au public avant la date de dépôt de la demande de brevet par une description écrite ou orale, un usage ou tout autre moyen. Est également considéré comme compris dans l'état de la technique le contenu de demandes de brevet belge et de demandes de brevet européen ou internationales désignant la Belgique antérieures à la date de la demande<sup>69</sup>. En d'autres termes, une invention ne peut être protégée que s'il s'agit d'un produit ou d'un procédé qui n'existaient pas publiquement avant qu'elle soit déposée par l'inventeur. Cette condition correspond à l'un des fondements du système des brevets : un monopole n'est accordé à un inventeur qu'à condition qu'il face avancer la technologie ou la science, en dévoilant publiquement le fruit de son travail d'invention.

Cette condition de nouveauté est généralement vérifiée avant qu'un brevet soit accordé. Il est cependant difficile, pour les institutions en charge de la délivrance des brevets, d'apprécier cette condition dans des les domaines technologiques où l'évolution est particulièrement rapide, ce qui est le cas du développement logiciel.

Une invention est considérée comme impliquant une **activité inventive** si, pour un homme du métier, elle ne découle pas d'une manière évidente de l'état de la technique<sup>70</sup>. Il ne suffit dès lors pas que l'invention soit nouvelle... il faut encore que

<sup>69</sup> Article 5 de la LBI, article 54 de la CEB.

<sup>70</sup> Article 6 de la LBI, article 56 de la CEB.

la solution technique qu'elle apporte fasse preuve d'inventivité, à savoir, d'une ingéniosité qui aille au-delà de ce qu'un homme du métier aurait fait de façon évidente. La logique est toujours la même : puisqu'il s'agit d'inciter le progrès technologique, on ne récompense pas ce que toute personne qui travaille dans le domaine aurait naturellement trouvé, mais bien les véritables efforts créatifs.

Une invention est considérée comme **susceptible d'application industrielle** si son objet peut être fabriqué ou utilisé dans tout genre d'industrie<sup>71</sup>. Autrement dit, les brevets ne peuvent protéger des inventions purement théoriques qui n'auraient aucune application pratique.

## II.2.d. Formalités

Contrairement aux droits d'auteur (qui s'acquiescent systématiquement, gratuitement et sans formalités), les brevets ne s'obtiennent que moyennant le dépôt d'une demande, le respect d'une procédure et le paiement de frais.

Le brevet n'est par ailleurs maintenu année par année que par le paiement d' « annuités ».

## II.2.e. Droits exclusifs

Le brevet confère le droit exclusif d'autoriser ou d'interdire à tout tiers<sup>72</sup> :

- la fabrication, l'offre, la mise dans le commerce, l'utilisation, l'importation ou la détention aux fins précitées du produit objet du brevet;
- l'utilisation d'un procédé objet du brevet ou, lorsque le tiers sait ou lorsque les circonstances rendent évident que l'utilisation du procédé est interdite sans le consentement du titulaire du brevet, l'offre de son utilisation sur le territoire belge;
- l'offre, la mise dans le commerce ou l'utilisation ou bien l'importation ou la détention aux fins précitées du produit obtenu directement par le procédé objet du brevet ;
- et dans une certaine mesure, la livraison à une personne autre que celles habilitées à exploiter l'invention brevetée des moyens de mise en œuvre de cette invention se rapportant à un élément essentiel de celle-ci, lorsque l'on sait ou lorsque les circonstances rendent évident que ces moyens sont aptes et destinés à cette mise en œuvre.

<sup>71</sup> Article 7 de la LBI, article 57 de la CEB.

<sup>72</sup> Article 27 de la LBI.

Les droits conférés par le brevet ne s'étendent cependant pas, entre autres, aux actes accomplis dans un cadre privé et à des fins non commerciales, ou aux actes accomplis à des fins scientifiques sur et/ou avec l'objet de l'invention brevetée<sup>73</sup>.

## II.2.f. Titularité du brevet.

Le droit au brevet européen appartient à l'inventeur ou à son ayant cause<sup>74</sup>.

Selon la CEB, Si l'inventeur est un employé, le droit au brevet européen est défini selon le droit de l'Etat dans lequel l'employé exerce son activité principale<sup>75</sup>.

La loi belge ne donne aucune précision quant aux inventions effectuées par les employés : ces questions devraient dès lors faire l'objet de clauses contractuelles.

En l'absence de clauses contractuelles, la jurisprudence fait une distinction entre les inventions de service, les inventions mixtes et les inventions libres.

Les inventions « de service » sont les inventions réalisées en exécution de la mission créative qui lui a été confiée de manière explicite ou implicite par son employeur. Ces inventions sont considérées comme étant d'office la propriété de l'employeur<sup>76</sup>.

Les inventions « libres » sont les inventions réalisées par l'employé en dehors de son activité comme employé et avec ses propres moyens. Ces inventions sont considérées comme étant la seule propriété de l'employé<sup>77</sup>.

Les inventions « mixtes » sont les inventions qui ne relèvent pas de l'exécution de sa mission par l'employé mais qui ont été réalisées grâce à des moyens matériels ou intellectuels procurés par l'employeur. La doctrine et la jurisprudence sont partagées quant à la solution à donner dans pareils cas<sup>78</sup>. Il appartiendra au juge de trancher le litige en fonction des caractéristiques du cas d'espèce.

Si plusieurs personnes ont réalisé l'invention indépendamment l'une de l'autre, le droit au brevet européen appartient à celle dont la demande de brevet européen a la date de dépôt la plus ancienne<sup>79</sup>.

Dans le cadre de la procédure de demande, le demandeur est réputé habilité à exercer le droit au brevet européen<sup>80</sup>.

---

<sup>73</sup> Article 28 de la LBI.

<sup>74</sup> Article 8 de la LBI, article 60 de la CEB.

<sup>75</sup> Article 60 de la CEB.

<sup>76</sup> M. BUYDENS, *op. cit.*, p.88 et s.

<sup>77</sup> M. BUYDENS, *op. cit.*, p.91 et s.

<sup>78</sup> M. BUYDENS, *op. cit.*, p.90 et s.

<sup>79</sup> Article 8 de la LBI, article 60 de la CEB.

<sup>80</sup> Article 8 de la LBI, article 60 de la CEB.

## II.2.g. Durée

La durée normale d'un brevet est de 20 ans à partir de la date du dépôt de la demande<sup>81</sup>.

---

<sup>81</sup> Art. 39 de la LBI, article 63 de la CEB.

## II.3. Droit des marques

Un tour d'horizon des droits intellectuels relatifs aux logiciels libre n'aurait pas été complet sans invoquer les droits de marques, qui concernent de plus en plus les noms des logiciels, des projets de développement et/ou des fournisseurs de services.

Par exemple les signes « Linux », « Red Hat », « Mozilla », « Apache », « Suze », « KDE », « Mandrake »,... de même que leurs logos (images de pingouins, dinosaures, chapeaux rouges, etc.) font l'objet de droits de marque.

Au contraire des droits d'auteur, lesquels sont donnés en licence de façon très permissive, les droits de marques ne font généralement pas l'objet de permissions quelconques.

Au contraire, certaines licences mentionnent spécifiquement que les autorisations d'usage concédées sur le code des logiciels ne s'étendent pas à l'usage de leurs marques, et qu'il est dès lors interdit d'utiliser les marques en question sans autorisation explicite de leurs détenteurs.

L'économie du « libre » étant essentiellement basée sur le service, on comprendra que les marques, identifiant les fournisseurs de services, aient une importance commerciale particulière et soient gardées sous contrôle exclusif.

### II.3.a. Plusieurs systèmes et bases légales multiples

Tout comme le droit des brevets, le droit des marques est une matière complexe et réglementée à différents niveaux. La marque est également un titre qui est décerné par une institution spécifique pour un territoire déterminé, et il est également possible d'obtenir des marques par plusieurs voies différentes.

Au niveau local, la Belgique, le Luxembourg et les Pays-Bas se sont réunis afin de créer un système uniforme « Benelux ». La [convention « Benelux »](#) en matière de propriété intellectuelle instaura la « marque Benelux », valable dans l'ensemble du territoire Benelux, et administrée par l'office Benelux de la propriété intellectuelle ([l'OBPI](#)).

Une marque communautaire couvrant l'ensemble du territoire de l'union européenne et dont le régime est instauré par le [règlement CE n°207/2009](#), peut être délivrée par l'office d'harmonisation du marché intérieur (l'[OHMI](#)).

Une procédure internationale de dépôt est également mise en place par le biais du [système de Madrid](#), administré par l'office mondial de la propriété intellectuelle (l'[OMPI](#)).

Etant donné la multiplicité des systèmes et la complexité de la matière, la présente étude n'aura pour seul objectif que d'en exposer les grands principes communs en Europe.

### II.3.b. Notions de base

La marque est un signe distinctif qui permet **d'identifier la provenance** d'un produit ou d'un service : l'objectif de la marque est que toute personne achetant un produit ou un service d'une marque déterminée soit assurée qu'un autre service ou produit portant la même marque provienne du même fabricant ou du même fournisseur de services.

Elle a également une **fonction publicitaire**, dans la mesure où une personne qui a été satisfaite par un produit ou un service d'une marque déterminée aura tendance à rechercher le même produit et/ou service et à le conseiller.

Le droit de marque est dès lors un droit qui donne à son titulaire l'**exclusivité de l'usage de la marque**, dans la vie des affaires, pour des produits ou des services spécifiques.

Afin de remplir sa fonction, la marque doit avoir des propriétés **distinctives**, c'est-à-dire, entre autres, ne pas être constituée d'un mot (signe) ou d'un ensemble de mots (signes) susceptibles de décrire le service, le produits ou leurs qualités intrinsèques, ou susceptibles d'être perçus comme remplissant une autre fonction par le public concerné (fonction décorative par exemple).

Les législations relatives aux droits de marques définissent une série de critères et de **conditions** qu'une marque doit respecter afin d'être valide.

Une marque s'obtient par un **dépôt** auprès d'un office déterminé, ce qui implique des démarches administratives et le paiement de frais.

Une marque ne peut se déposer que dans des **classes de produits ou de services** spécifiques. Le monopole qui est accordé au détenteur de la marque ne l'est que pour certains produits ou certains services, et que pour autant que la marque soit **effectivement utilisée** pour distinguer ces produits et services.

### II.3.c. Droits exclusifs et portée de la marque

En général, la marque communautaire confère à son titulaire un droit exclusif d'interdire à tout tiers de faire usage dans la vie des affaires:

- d'un signe identique à la marque pour des produits ou des services identiques à ceux pour lesquels celle-ci est enregistrée;
- d'un signe similaire à la marque pour des produits ou services similaires, lorsque ces similarités sont source d'un risque de confusion dans l'esprit du public;
- d'un signe identique ou similaire à la marque pour des produits ou des services qui ne sont pas similaires à ceux pour lesquels la marque est enregistrée, lorsque celle-ci jouit d'une renommée dans le territoire couvert et que l'usage du signe se fait sans juste motif et qu'il tire indûment profit du caractère distinctif ou de la renommée de la marque, ou leur porte préjudice.

Il peut notamment être interdit d'apposer le signe sur les produits ou sur leur conditionnement, d'offrir les produits ou de les mettre dans le commerce ou de les détenir à ces fins, d'offrir ou de fournir des services sous le signe, d'importer ou d'exporter les produits sous le signe, d'utiliser le signe dans les papiers d'affaires et la publicité.

### II.3.d. Titularité

Les droits de marque sont détenus par la personne qui l'a déposée.

### II.3.e. Durée

La durée originale d'une marque est de 10 ans, mais elle est indéfiniment renouvelable, à chaque fois pour une période équivalente.

## II.4. Tableau comparatif

	<b>Droit d'auteur</b>	<b>Brevets</b>	<b>Marques</b>
<b>Objet de la protection</b>	Les œuvres littéraires et artistiques  (auxquelles sont assimilés les programmes d'ordinateur).	Les inventions, c'est-à-dire les solutions techniques à des problèmes techniques  (inventions mises en œuvre par programme informatique).	Les signes (mots, logos, etc.) permettant de distinguer un produit ou un service d'un autre produit ou service.
<b>Etendue de la protection</b>	Les éléments originaux de la forme.  (le code)	Les procédés techniques  (les fonctionnalités)	L'usage du signe pour distinguer un ou plusieurs produits ou services déterminés
<b>Conditions</b>	- mise en forme - originalité	- nouveauté - activité inventive - application industrielle	- signe pouvant être reproduit graphiquement. - signe distinctif - disponibilité du signe (pas identique ou similaire à un signe utilisé pour produit ou service identique ou similaire) - restrictions diverses
<b>Durée</b>	70 après la mort de l'auteur	20 ans maximum	10 ans - renouvelable indéfiniment

### III. Aspects contractuels

Les licences FOSS sont avant tout des licences de droit d'auteur, mais comprennent également, entre autres, certaines clauses relatives aux marques et aux brevets, des clauses imposant certaines obligations au licencié, ainsi que des clauses relatives aux garanties et à la responsabilité des développeurs.

Malgré le fait que cette qualification peut être contestée dans certains pays, les licences FOSS seront immanquablement qualifiées de contrat en droit belge.

La présente section a pour objectif d'analyser à grands traits les aspects contractuels des licences FOSS.

#### III.1. Licence FOSS : contrat d'autorisation donnée sur la base de droits intellectuels

##### III.1.a. Les licences FOSS sont des contrats

Nous avons vu dans le cadre de la section II que les logiciels étaient avant tout protégés par les droits d'auteur, qui sont des droits exclusifs d'autoriser ou d'interdire certains actes. L'objectif principal des licences FOSS est de gérer ces droits d'auteur de façon à autoriser l'usage et la modification des logiciels, et d'établir les termes et conditions en contrepartie de cette autorisation.

Contrairement à certaines idées qui circulent au sein des communautés du libre et largement inspirées par des débats juridiques anglo-saxons<sup>82</sup>, les licences FOSS ne peuvent s'analyser, en droit belge, que comme des contrats<sup>83</sup> passés entre un titulaire de droits (le donneur de licence) et un bénéficiaire (licencié).

---

<sup>82</sup> En effet, on constate qu'un débat jurisprudentiel américain (sur le fait de savoir si certaines licences open source (dont la GPL) sont des contrats) a eu tendance à confondre les développeurs européens. Si la question à certains intérêts dans les systèmes de *common law*, elle a moins d'incidence dans les pays de tradition « civiliste » dont fait partie la Belgique.

<sup>83</sup> Y. COOL, "Aspects contractuels des licences de logiciels libres : les obligations de la liberté", in *Les logiciels libres face au droit*, Bruxelles, Bruylant, 2005, p.137 et s. C'est également le cas en France (H. BITAN, *Protection et contrefaçon des logiciels et des bases de données*, Lamy, 2006, p. 141), aux Pays-Bas (E.P.M. THOLE & W. SEINEN, « Open source-softwarelicenties : een civielrechtelijke analyse », *Computer Recht*, 2004, n°5, p.221 et s.), voire dans l'ensemble de l'Europe (A. SIEDLECKA-VAN RUMST, « Validity Issues of Open Sources Licences : the EU Solution », *MUJLT*, 2008, n°2, p.65.).

Parmi les différentes sources d'obligations<sup>84</sup>, le contrat est en effet la seule catégorie juridique qui correspond à une situation dans laquelle deux personnes conviennent de faire naître des obligations dans le chef d'une de ces personnes (contrat unilatéral) ou de ces deux personnes (contrat synallagmatique).

En l'occurrence les licences FOSS s'analyseront souvent comme des contrats synallagmatiques, dans la mesure où elles font naître des obligations dans le chef des deux parties<sup>85</sup> :

- **Le donneur de licence** s'engage normalement à concéder des droits couverts par ses prérogatives exclusives :
  - o droit d'utiliser le logiciel,
  - o droit de le reproduire,
  - o droit de le redistribuer et/ou de le communiquer,
  - o droit de modifier le logiciel et
  - o droit de redistribuer et/ou communiquer la version modifiée.

Certaines licences prévoient également l'obligation explicite dans le chef du donneur de licence de communiquer les codes sources des logiciels (même si cette obligation va de soi)<sup>86</sup>.

- Les obligations du **Licencié** sont multiples et varient en fonction des licences. Certaines obligations courantes sont reprises ci-dessous.

### III.1.b. Règles contractuelles spécifiques en matière de licences de droit d'auteur

La cession (ce terme regroupe aussi bien le transfert que la licence) de droits d'auteur est soumise à un certain formalisme établi par le législateur afin de protéger les auteurs. De même, des règles d'interprétation favorables aux auteurs ont été instaurées. Ces règles sont générales et sont reprises à l'article 3 de la LDA. La LPO restant muette à cet égard, la doctrine considère qu'elles sont également d'application en matière de logiciels<sup>87</sup>.

On notera toute fois que ces règles protectrices des auteurs ne bénéficient qu'aux auteurs originaires (à savoir, les personnes physiques ayant créé les œuvres). Elles

---

<sup>84</sup> Qui comprennent les contrat, les quasi-contrat, les quasi-délits, les délits et les actes unilatéraux.

<sup>85</sup> Y. COOL, *op. cit.*, p.137 et s.; G. JOCQUEL, *Les licences libres en droit français : analyse comparée CeCILL – Gnu GPL*, septembre 2004, p.15, disponible sur le site <http://www.juriscom.net>.

<sup>86</sup> Y. COOL, *op. cit.*, p.141.

<sup>87</sup> A. STROWEL & E. DERCLAYE, *op. cit.* p. 206 et s.

ne devraient donc pas bénéficier aux cessionnaires des droits (et donc, entre autres, aux sociétés)<sup>88</sup>.

### III.1.b.1. Ecrit

L'article 3 §1, al.1 de la LDA prévoit que les droits patrimoniaux sont mobiliers, cessibles et transmissibles, en tout ou en partie, conformément aux règles du code civil. Ils peuvent notamment faire l'objet d'une aliénation ou d'une licence simple ou exclusive.

En matière de FOSS, il s'agit bien entendu de licences « simples », autrement dit « non-exclusives ».

L'article 3 §1, al.2 de la LDA prévoit qu'à l'égard de l'auteur, tous les contrats se prouvent par écrit. Cette exigence d'écrit n'est que probatoire : ce n'est que lorsqu'il est demandé au licencié de prouver l'existence et le contenu de sa licence que cette preuve devra être apportée par la production d'un « écrit ». L'exigence d'un « écrit » n'impose pas qu'une licence doive prendre la forme d'un contrat daté et signé de la main des deux parties. La doctrine s'accorde à dire que tout type d'écrit peut faire office, en ce compris l'échange de courrier, des conditions générales, un bon de commande...

Transposé dans le monde du « libre » il semble dès lors suffisant, pour que cette condition soit remplie, que la licence soit attachée à un logiciel sous forme de fichier, ou même, qu'il y soit fait clairement référence ainsi qu'à la page web dans laquelle les termes et conditions de la licence sont repris. (Voy. Cependant la partie consacrée à la validité des contrats).

### III.1.b.2. Mentions obligatoires

L'article 3, §1, al.4 de la LDA prévoit que, pour chaque mode d'exploitation, la rémunération de l'auteur, l'étendue et la durée de la cession doivent être déterminées expressément.

Le niveau de précision requis pour la mention des **modes d'exploitation** fait l'objet de certaines controverses. Les travaux préparatoires de la LDA précisent que cette notion de « mode d'exploitation » ne doit pas s'interpréter dans un sens trop restrictif

<sup>88</sup> Voir Civ. Bruxelles (cess.), 25 janvier 2002, A&M, 2004/4, p. 333 et s.

mais de manière opérante, et qu'il n'est pas nécessaire que chaque mode distinct d'exploitation soit détaillé<sup>89</sup>.

Cependant, la doctrine s'accorde sur le fait qu'au minimum chaque droit patrimonial visé doit être mentionné. Il s'agit également, en fonction des secteurs, de désigner les supports et la nature de l'exploitation<sup>90</sup>.

En ce qui concerne les FOSS, les termes employés pour désigner ce qui est autorisé sont généralement larges, et peu d'indications quant aux supports ou à l'exploitation sont présentes. Ceci peut s'expliquer par la volonté de permissivité et de liberté qui accompagne ces licences.

Lorsque cette obligation est jointe à la règle d'interprétation stricte des contrats en faveur des auteurs (cfr. infra), l'absence de toute référence au « droit de communication au public » ou à tout équivalent dans certaines licences américaines inquiète plus, dans la mesure où la diffusion des logiciels se fait communément, à l'heure actuelle, par la mise à disposition des codes sur Internet. Pour pallier cette carence, il s'agirait, le cas échéant, d'invoquer les « pratiques du secteur ».

La **rémunération** peut être unique, forfaitaire, en fonction de l'usage, etc... Elle peut également être nulle.

Le caractère « gratuit » des licences FOSS est généralement clairement exprimé dans leurs énoncés.

L'**étendue** vise en fait l'« étendue géographique », soit le territoire sur lequel la licence sera d'application.

L'aspect « mondial » des licences FOSS est généralement mentionné.

La licence peut être limitée à une certaine **durée**.

Dans le cas des logiciels FOSS, cette durée est généralement définie comme étant « la durée des droits d'auteur ».

En cas d'absence d'une mention légale, la loi reste muette quant aux sanctions. Selon certains, la sanction est la nullité de la convention<sup>91</sup>, pour d'autre la nullité est

<sup>89</sup> A. STROWEL & E. DERCLAYE, *op. cit.* p. 108 et s.

<sup>90</sup> F. DE PATOUL, *op. cit.*, p.123.

excessive et la sanction adéquate équivaut à l'application de la règle d'interprétation stricte<sup>92</sup> (cfr. Infra).

### **III.1.b.3. Interprétation stricte en faveur de l'auteur**

L'article 3, §1, al.3 de la LDA prévoit que les dispositions contractuelles relatives au droit d'auteur et à ses modes d'exploitation sont de stricte interprétation.

On ajoutera cependant qu'il n'y a besoin d'interpréter une clause que quand celle-ci n'est pas claire. Cette règle est d'application en matière de licences FOSS, mais aura une portée limitée, dans la mesure où ces dernières sont rédigées de façon assez large et « englobante ».

### **III.1.b.4. Obligation d'exploitation conformément aux usages honnêtes**

Selon l'article 3, §1, al.5 de la LDA, le cessionnaire est tenu d'assurer l'exploitation de l'œuvre conformément aux usages honnêtes de la profession.

Cette règle, qui a davantage de relevance en matière de transferts de droits ou de licences exclusives, n'a pas beaucoup d'intérêt dans le cadre de l'octroi systématique de licences non-exclusive relatives à des logiciels.

Cette règle ne devrait avoir que peu d'incidence en matière de FOSS.

### **III.1.b.5. Nullité des cessions concernant des formes d'exploitation encore inconnues**

L'article 3, §1, al.6 de la LDA prévoit que nonobstant toute disposition contraire, la cession des droits concernant des formes d'exploitation encore inconnues est nulle. C'est davantage la prévisibilité d'un mode d'exploitation qui est visé plus que son caractère « connu » ou non.

---

<sup>91</sup> F. DE PATOUL, *op. cit.*, p.127, H. BITAN, *op.cit.*, p.151.

<sup>92</sup> A. STROWEL & E. DERCLAYE, *op. cit.* p. 112 et s.

Cette règle est théoriquement applicable en matière de FOSS. Il est bien évidemment peu aisé d'imaginer les formes d'exploitations de logiciels encore inconnues (non prévisibles) à l'heure actuelle.

### **III.1.b.6. Cessions de droits sur des œuvres futures**

L'article 3, § 2 de la LDA prévoit que la cession des droits patrimoniaux relatifs à des œuvres futures n'est valable que pour un temps limité et pour autant que les genres des œuvres sur lesquelles porte la cession soient déterminés.

Cette règle n'a également pas beaucoup d'intérêt en matière de FOSS, dans la mesure où les logiciels sont créés avant qu'ils ne soient distribués sous licence.

### **III.1.b.7. Contrats de travail et de commande**

L'article 3, § 3 de la LDA prévoit certaines dispositions en cas de contrat de travail ou de contrat de commande, notamment certaines exceptions aux règles contractuelles exposées ci-dessus :

*« Lorsque des œuvres sont créées par un auteur en exécution d'un contrat de **travail** ou d'un statut, les droits patrimoniaux peuvent être cédés à l'employeur pour autant que la cession des droits soit expressément prévue et que la création de l'œuvre entre dans le champ du contrat ou du statut.*

*Lorsque des œuvres sont créées par un auteur en exécution d'un **contrat de commande**, les droits patrimoniaux peuvent être cédés à celui qui a passé la commande pour autant que l'activité de ce dernier relève de l'industrie non culturelle ou de la publicité, que l'œuvre soit destinée à cette activité et que la cession des droits soit expressément prévue.*

*Dans ces cas, le § 1er, alinéas 4 à 6, et le § 2 ne s'appliquent pas.*

*La clause qui confère au cessionnaire d'un droit d'auteur le droit d'exploiter l'œuvre sous une forme inconnue à la date du contrat ou de l'engagement sous statut doit être expresse et stipuler une participation au profit généré par cette exploitation.*

*Des accords collectifs peuvent déterminer l'étendue et les modalités du transfert. »*

En ce qui concerne les contrats de travail, référence est faite à la section consacrée à la titularité des droits d'auteur (cfr. supra).

Nous remarquerons qu'en cas de contrat de commande dans le cadre d'une activité qui relève de la publicité ou de l'industrie non culturelle, le régime de formalité est assoupli : les règles concernant les mentions obligatoires, l'obligation d'exploitation et la cession des droits concernant des formes d'exploitation encore inconnue ne sont pas applicables.

### **III.1.b.8. Considérations finales**

Les problèmes liés au non-respect de certaines règles de formalisme dans certaines licences sont souvent soulevés comme contribuant à l'insécurité juridique des licences FOSS<sup>93</sup>. Si ces critiques sont fondées, il s'agit cependant de souligner l'aspect particulier des licences libres (licences permissives offertes à quiconque en accepte ses termes), l'esprit dans lequel elles sont rédigées et concédées et le régime juridique qui a influencé leur rédaction (beaucoup de licences étant inspirées du droit américain, qui ne prévoit pas les mêmes règles de formalisme).

D'autre part, la progression du développement de logiciels open source en entreprise aura pour conséquences une diminution de l'applicabilité de ces règles au profit de l'application des règles contractuelles de droit commun.

### **III.1.c. Règles spécifiques en matière de licences de brevets**

Il existe beaucoup moins de règles contractuelles spécifiques en matière de brevet : en général, ce sont les règles de droit commun qui s'appliquent. La raison est sans doute que le législateur a estimé que le titulaire d'un brevet n'était pas en position de faiblesse dans le cadre d'une négociation contractuelle, et ne devait dès lors pas être protégé par un formalisme quelconque.

Cependant, il paraît intéressant d'analyser une règle spécifique relative aux licences de brevets, à savoir l'obligation de notifier les licences aux offices qui les ont

---

<sup>93</sup> H. BITAN, *op.cit.*, Lamy, 2006, p. 151 et s. ; F. DE PATOUL, *op. cit.*, p.128 ; C. ROJINSKY & V. GRYNBAUM, "Les licences libres en droit français", *Propriétés Intellectuelles.*, 2002, p.28 et s.

décernés. En effet, les offices ont pour mission de conserver des registres avec l'ensemble des brevets en vigueur y d'y ajouter les licences concédées à des licenciés.

Cette règle a pour conséquence que tant qu'une licence n'est pas inscrite dans les registres de l'office, elle n'est opposable ni à cette dernière, ni aux tiers<sup>94</sup>.

On peut s'interroger sur l'application de ces règles en matière de licences FOSS.

Il est de plus en plus courant de voir des licences FOSS comportant des clauses relatives à des licences octroyées sur tous les brevets que pourraient détenir le licencié sur les technologies implémentées par le logiciel sur lequel porte la licence FOSS. Il est évident que ces licences ne font, à l'heure actuelle, jamais l'objet d'inscriptions dans les registres des offices de brevet (les brevets en question étant hypothétiques et non identifiés, et les licences étant concédées en grand nombre).

Si pareilles clauses permettront généralement de protéger les licenciés contre toute action intentée à leur encontre par le donneur de licence sur la base de ces brevets, elle deviendra « inopposable » à tout tiers qui pourrait les acquérir par la suite.

## III.2. Autres aspects contractuels

### III.2.a. Obligations du licencié

Afin de profiter des termes très permissifs des licences FOSS, le licencié doit malgré tout respecter certaines obligations.

L'étendue et le contenu de celles-ci varient d'une licence à une autre : il est dès lors très difficile d'établir un relevé des obligations du licencié sans faire une analyse licence par licence.

#### III.2.a.1. Obligations récurrentes

---

<sup>94</sup> Voir article 45 de la LBI

Certaines obligations sont imposées au licencié de façon récurrente.

Toutes les licences FOSS imposent au licencié le respect des « notices » et l'interdiction d'effacer ces-dernières.

Cela comprend généralement le respect des mentions de droit d'auteur (nom et année de l'auteur) ainsi que des avertissements relatifs au refus de garantie et à la limitation de responsabilité.

Les licences imposent également généralement au licencié de faire accompagner le logiciel du texte de la licence.

### **III.2.a.2. Les clauses qui font la différence**

#### **III.2.a.2.1. Clause « copyleft »**

L'une des distinctions fondamentales qui peuvent être effectuées entre les licences est sans doute leur caractère copyleft ou non-copyleft (les clauses copyleft se déclinant par ailleurs en une multiplicité de formes aux effets et à l'étendue différents).

L'effet copyleft d'une licence FOSS est celui qui, en général, impose au licencié qui désire redistribuer une version du logiciel modifiée ou améliorée par ses soins de le faire sous la même licence FOSS.

Cet effet a donc pour principale conséquence qu'un logiciel FOSS reste FOSS, même dans ses formes modifiées. Elle a également pour conséquence que le logiciel modifié ne pourra jamais être distribué sous une autre licence, et ne pourra donc, par la suite, être exploité en mode « propriétaire » par le licencié.

On insistera sur le fait que l'effet copyleft des licences de ce type ne se déclenche, classiquement, que lorsqu'il y a « distribution » du logiciel modifié. Autrement dit, ce n'est que lorsque le licencié décide de distribuer son code qu'il est tenu de respecter la clause copyleft. A contrario, un licencié qui se contente de modifier un logiciel FOSS sous licence copyleft pour son usage personnel (ou interne à l'entreprise<sup>95</sup>) n'est pas obligé de redistribuer sa version modifiée. L'une des questions épineuses sur laquelle il s'agira de se pencher davantage est celle qui a trait à déterminer les cas où il y a effectivement distribution du logiciel des situations où le logiciel n'est

---

<sup>95</sup> B. EGRET & Y. KAPLUN, «La nécessaire interprétation des licences relatives aux logiciels libres», *R.L.D.I.*, 2006, n°12, p.52 ; J.P. TRIAILLE, « Licences « open source » et contrats avec les auteurs et les distributeurs », *R.D.T.I.*, 2005, p.61.

utilisé que de manière interne. En outre, certaines, licences<sup>96</sup> étendent l'effet copyleft aux technologies dites « Software as a Service », dans le cadre desquelles les logiciels ne sont pas distribués mais mis à disposition à distance<sup>97</sup>.

La question de « distribution » en tant qu'élément déclencheur de l'effet copyleft devrait faire l'objet d'une analyse spécifique.

Signalons également que l'effet copyleft et, entre autres, son étendue varient d'une licence à l'autre. Certaines licences sont rédigées afin que l'effet copyleft couvre le plus de code possible (effet copyleft « fort »). D'autres licences ont une clause copyleft rédigée afin que l'effet ne s'étende qu'à certaines parties du logiciel. Généralement, l'étendue de l'effet copyleft est définie par des termes techniques ou se définit en fonction de techniques de programmation utilisées (extension de l'effet copyleft au « linkage dynamique », limitation de l'effet copyleft « par fichier », etc.)

Les questions liées à la classification des effets copyleft, l'appréciation de leur conformité avec les règles de droit d'auteur ou d'autres législations, et l'influence de la technique en la matière devraient faire l'objet d'une analyse spécifique.

### **III.2.a.2.2. Liste d'autres clauses fréquentes et qui différencient les licences les unes des autres**

Sous ce point sont répertoriées quelques autres clauses « type » qui distinguent les licences les unes des autres :

- Lorsque le logiciel a été modifié, certaines licences imposent de le mentionner, en précisant parfois les informations à fournir.
- Certaines licences comportent des clauses spécifiant la façon dont les codes sources doivent être mis à disposition des licenciés.
- Certaines licences règlent la façon dont les codes exécutables peuvent être distribués.
- Certaines clauses portent sur les droits des marques apposées sur les logiciels.

---

<sup>96</sup> Des licences de type « Affero », tels que l' « Affero GPL ».

<sup>97</sup> B. BEUVICK & M. WESTERIK, «Software as a Service : waar Open Source Software ophoudt open te zijn», *Tijdschrift voor Internetsrecht*, 2008, n°6, p. 152 et s.

- Certaines licences comprennent des clauses relatives à l'éventuelle protection des logiciels par brevets.
- Certaines licences déterminent les conséquences découlant du non-respect de la licence ou de l'une de ses clauses.
- Certaines licences comportent des clauses déterminant la loi applicable ou le juge compétent.
- Certaines clauses spécifient les logiciels avec lesquels le logiciel couvert peut être combiné ou fusionné en fonction de la licence de ceux-ci (clauses de « compatibilité »).
- ...

La question de la compatibilité des licences libres devrait faire l'objet d'une étude spécifique.

## III.2.b. Validité du contrat et de certaines clauses

S'agissant de contrats, beaucoup de questions se sont posées sur les aspects techniques de la conclusion des contrats de licence FOSS, et entre autres, sur leur validité.

La validité de du contrat ou de certaines clauses peut être influencée par la qualité de consommateur de l'utilisateur-licencié, qui est protégé, par exemple, contre les clauses abusives, la non-conformité des produits, les pratiques abusives en matière de contrats à distances, etc...

Dans cadre du présent rapport, il sera présumé que l'utilisateur du logiciel FOSS n'est pas un consommateur. Les règles spécifiques en matière de droit de la consommation ne seront pas abordées (mais devraient sans doute l'être dans le cadre d'une étude spécifique).

### III.2.b.1. Langue

La majorité des licences FOSS sont écrites en anglais. A cet égard, le droit commun est d'application : la langue n'est pas un obstacle en soi, mais une personne qui accepte un contrat doit normalement en comprendre les termes.

Si cette personne ne maîtrise pas la langue du contrat, il se peut qu'elle n'en comprenne pas le contenu et qu'elle commette dès lors une erreur.

En cas d'erreur, la validité d'un contrat ne peut être mise en question que lorsque cette erreur est déterminante, c'est-à-dire, lorsque la personne qui a commis l'erreur n'aurait pas contracté si elle avait réellement compris la teneur des éléments sur lesquels a porté l'erreur.

Par ailleurs, cette action en annulation n'est possible que si l'erreur est excusable. Or la doctrine et la jurisprudence ont tendance à estimer qu'est inexcusable le fait d'accepter sans réserve un contrat dans une langue qu'on ne maîtrise pas.<sup>98</sup>

En résumé, le fait que les licences FOSS soient rédigées en anglais ne devrait pas, en pratique, poser trop de problèmes de validité en Belgique, dans la mesure où une personne qui aura accepté une licence en anglais ne pourra pas par la suite tenter d'invoquer le fait qu'elle ne comprenait pas l'anglais afin d'annuler le contrat.

## **III.2.b.2. Acceptation**

### **III.2.b.2.1. Techniques de validation des licences**

Une autre question relative à la validité des licences FOSS est de savoir si le licencié a pu effectivement accepter les termes et conditions de la licence.

Traditionnellement, la meilleure façon de prouver qu'un contrat est accepté par une partie est d'en reprendre les termes dans un document écrit daté et signé par les cocontractants. Il est évident que pareille formalisme est impraticable dans un secteur comme le secteur du logiciel, ou tout usage nécessite licence, et dès lors, ou chaque installation de logiciel implique acceptation d'un contrat.

Plusieurs techniques ont été mises au point afin de s'assurer de l'acceptation de la licence par le licencié sans devoir passer à la signature d'un document.

La première technique qui fut inventée fut les licences « *shrinkwrap* », c'est-à-dire les licences qui faisaient découler leur acceptation de l'ouverture de l'emballage des supports de logiciels. Puis furent inventées les techniques de « *clickwrap* », consistant à faire défiler les termes de la licence à l'écran avant l'usage du logiciel, et de ne permettre ce-dernier que si l'utilisateur a formellement appuyé sur un bouton « j'accepte ». L'attention de l'utilisateur étant spécialement attirée sur le contrat de licence, et le passage à une étape postérieure dans l'installation et/ou l'utilisation du logiciel impliquant obligatoirement son acceptation, ces techniques sont

---

<sup>98</sup> Y. Cool, *op. cit.*, p. 167 et s.

généralement bien acceptées par la doctrine et la jurisprudence. C'est la raison pour laquelle ces systèmes sont très répandus dans la pratique.

La technique de « *browsewrap* » consiste à renvoyer l'utilisateur à un document contractuel qui se trouve généralement à un autre endroit (autre fichier ou page internet) que l'écran d'usage ou d'installation. Une marque d'approbation de la licence par l'utilisateur n'étant pas un passage obligatoire dans pareille situation, il devient plus difficile d'apporter la preuve de l'acceptation<sup>99</sup>.

On soulignera cependant qu'utiliser un logiciel en prétendant ne pas avoir accepté une licence ou en prétextant la confusion de licence sera assimilé à une violation de droit d'auteur.

En matière de licences FOSS, la licence du logiciel est généralement mentionnée dans les « headers » des codes sources. Soit elle y est intégralement reprise, soit il y est uniquement fait référence (par exemple : « ce logiciels est sous licence GPL v2.1 » - l'utilisateur étant présumé connaître la licence ou à en chercher le contenu sur internet). Aucun mécanisme d'acceptation n'étant généralement mis en place, pareille situation peut être assimilée à une situation de « *browsewrap* ».

Les logiciels libres distribués sous format « exécutables » sont quant à eux généralement pourvu d'un système de « *clickwrap* ».

Beaucoup de licences FOSS présument l'acceptation de la licence du seul fait de l'usage du logiciel, arguant que si le logiciel est utilisé sans acceptation, l'utilisateur se trouve alors en situation illégale. Le lien de cause à effet entre l'usage du logiciel et l'acceptation de la licence est cependant contestable<sup>100</sup>. On peut par exemple penser à des situations où un logiciel n'afficherait pas les bonnes références aux licences applicables, ou que l'information sur le régime légal d'un logiciel soit erroné ou effacé. Dans ce cas d'espèce, une personne pourrait utiliser un logiciel en pensant qu'une autre licence est d'application. Si pareille situation peut s'assimiler à une utilisation illégitime, on ne peut cependant pas conclure pour autant que l'utilisateur aurait en fait d'office accepté une licence dont il n'avait même pas connaissance.

### **III.2.b.2.2. L'acceptation des licences FOSS dans les entreprises**

Tout contrat passé avec une société doit normalement être signé par une personne dument habilitée à engager celle-ci. Généralement, un développeur n'a pas ce pouvoir au sein de son entreprise.

<sup>99</sup> CHR. DE PRETERE & H. DEKEYSER, "De totstandkoming en draagwijdte van open source-licenties", *Computer Recht*, 2004, n°5, p.217.

<sup>100</sup> J.P. TRIAILLE, *op. cit.*, p.53.

Dès lors, si un développeur a intégré du code FOSS dans un logiciel développé pour le compte de sa société, il n'est pas certain que cette dernière ait effectivement accepté les termes et conditions de la licence de ce code<sup>101</sup>.

A nouveau, à supposer que le développeur ait commis une faute et que la société n'accepte pas les termes de la licence FOSS, il va de soi que pareille situation est assimilable à une situation de contrefaçon.

Cependant, il existe des cas où une société aura davantage intérêt à nier l'acceptation d'une licence FOSS qu'à se voir forcée d'en remplir les obligations. Ce sera entre autres le cas lorsque la société en question développe du logiciel propriétaire et que le code FOSS inséré est distribué sous une licence à l'effet copyleft extensif. Dans ce cas, la dite société pourrait préférer cesser l'usage du code copyleft et faire face à une éventuelle demande en dommages et intérêts plutôt que redistribuer l'entièreté du logiciel sous licence FOSS.

### **III.2.b.3. Validité des clauses relatives aux garanties et à la responsabilité**

La question de la validité des clauses d'exonération de responsabilité et de non-garanties est souvent soulevée en matière de logiciels FOSS.

Soulignons que si le contrat de licence prévoit des clauses de non-garantie et d'exonération de responsabilité, cela n'empêche que des contrats spécifiques de garantie soient conclus par des fournisseurs de services avec leurs clients.

#### **III.2.b.3.1. Limitation de responsabilité**

Le raisonnement de départ est, en soi, légitime. Une licence sur le logiciel est donnée gratuitement, et celui-ci peut dès lors faire l'objet d'une diffusion gratuite : l'auteur, n'étant pas rémunéré, refuse d'assumer un quelconque risque au cas où le logiciel causerait des dommages.

La question est dès lors de savoir si le donneur de licence peut s'exonérer de toute responsabilité, et si cette clause tirera ses effets en toutes circonstances.

---

<sup>101</sup> J.P. TRIAILLE, *op. cit.*, p.53.

La validité et l'application des clauses de limitation de responsabilité sont généralement acceptées. Elles ne seront inopposables que dans certains cas précis.

Nul ne peut s'exonérer de son **dol**. Il y aura dol lorsque le développeur aura sciemment créé un code qui préjudiciera au licencié (un virus par exemple). Dans ce cas, aucune clause contractuelle ne peut couvrir pareil agissement, et le responsable sera toujours amené à réparer les dommages causés volontairement.

En matière de **fautes lourdes** (c'est-à-dire une faute particulièrement grossière issue d'une insouciance ou négligence graves, ou d'une violation à une obligation essentielle), le donneur de licence ne peut s'exonérer de sa responsabilité que si la faute lourde a été explicitement visée dans le contrat. Il s'agira dès lors de vérifier si ce cas est explicitement mentionné dans la clause limitant la responsabilité du donneur de licence.

Certains régimes spécifiques de responsabilité sont imposés par la loi et ne peuvent pas être écartés : c'est le cas de la **responsabilité des produits défectueux** (qui impose au producteur l'obligation d'indemniser les dommages causés aux personnes et, dans une certaine mesure, aux biens). Un logiciel pourrait effectivement être « embarqué » dans un appareil qui sera considéré comme produit au sens de la loi<sup>102</sup>. Par ailleurs, il est admis que cette législation, issue d'une directive européenne<sup>103</sup>, devrait s'appliquer aux logiciels en tant que tels<sup>104</sup>, même si la loi belge n'est actuellement pas conforme à la directive en question<sup>105</sup>.

Par ailleurs, une clause d'exonération de responsabilité ne peut porter atteinte à une obligation essentielle du contrat. Dans le cas des licences FOSS, cette situation ne devrait pas se présenter, l'obligation essentielle du contrat étant d'octroyer une autorisation sur la base de droits intellectuels exclusifs<sup>106</sup>.

### **III.2.b.3.2. Absence de garantie**

La limitation de toute garantie est également une possibilité offerte à tout fournisseur mais qui est également soumise à certaines limites, et entre autres, certaines garanties imposées par la loi, à supposer qu'elles soient applicables dans le cas d'espèce.

---

<sup>102</sup> S. VAN CAMP, "Open Source Licenties : de ondraaglijke lichtheid van een concept", *R.D.C.*, 2006, n°5, p. 506.

<sup>103</sup> Directive 85/374/CEE du 25 juillet 1985 relative au rapprochement des dispositions législatives, réglementaires et administratives des États membres en matière de responsabilité du fait des produits défectueux, *J.O.C.E.*, L 210, 7 août 1985, p.29.

<sup>104</sup> C. ROJINSKY & V. GRYNBAUM, *op.cit.*, p.36

<sup>105</sup> La loi belge n'appliquant ce régime qu'à "tout bien meuble corporel". Voir par exemple Y. COOL., *op.cit.*, p.185

<sup>106</sup> Y. COOL., *op.cit.*, p.187

La **garantie contre les vices cachés**<sup>107</sup> est parfois invoquée en matière logicielle : outre le fait qu'elle permet la résiliation du contrat ou la réduction du prix, elle permet également de réclamer tous les dommages causés par le vice lorsque le vendeur était au courant de sa présence. Il s'agit d'une disposition qui n'est, selon le texte de loi, applicable qu'en matière de vente. La jurisprudence a cependant étendu la garantie aux contrats de services. La qualification du contrat de licence de logiciel a toujours fait l'objet de controverses. Cependant, la qualification de contrat de vente ou de contrat de service pour ce type de contrats paraît de moins en moins évoquée. L'application de la garantie pour vices cachés en matière de licence FOSS est dès lors peu envisageable.

Le code civil prévoit également une **garantie d'éviction**, selon laquelle un vendeur doit garantir l'acheteur contre toute action de tiers troublant sa jouissance paisible du bien en prétendant avoir des droits sur ce-dernier. Cette garantie n'est cependant également applicable qu'en cas de vente. Pareille garantie ne s'appliquera donc pas aux licences FOSS, sauf si elle est explicitement prévue.

---

<sup>107</sup> Art. 1641 et s. du code civil.

## IV. Points à approfondir

L'étude sur l'état de l'art a permis de dégager une série de thématiques qui nécessiteraient des analyses complémentaires.

Elles sont reprises ci-dessous :

- Les notions d' « œuvre dérivée » et d' « œuvre basée sur un programme précédent » sont des notions centrales auxquelles de nombreuses licences font appel, sans pour autant que leurs contours soient légalement bien définis, ni qu'il y ait unanimité quant à leur interprétation. Une étude sur la notion d'œuvre dérivée, et sur le concept de licence permettant la création de pareilles œuvres, paraît nécessaire.
- Le concept de « distribution », auquel s'attachent d'importants effets, dont le déclenchement d'effets copyleft, devrait être analysé, ainsi que son usage dans différentes licences.
- L'effet « copyleft » devrait également faire d'une analyse comparative, celui-ci étant plus ou moins « fort » ou imposant différentes conditions en fonction des licences.
- La question de la validité des exonérations de responsabilité et de limitation de garantie devraient être plus précisément examinées, entre autres dans le rapport « Business to Consumer ».

Le problème de compatibilité de licences est également une problématique récurrente. Elle devrait faire l'objet d'études afin de dégager une méthodologie de gestion et de résolution des conflits d'obligations entre licences.