

PAR PASCAL AGOSTI



docteur en droit,
avocat associé, cabinet
Caprioli & Associés

Le logiciel **libre**, autre modèle de propriété **intellectuelle**

ET JEAN-SÉVERIN LAIR (87)



sous-directeur
des systèmes
d'information,
ministère
de la Culture et de
la Communication

Créés par des communautés fortes de développeurs et d'utilisateurs, les logiciels libres sont dans une dynamique d'amélioration continue. La réussite de cette approche coopérative a permis l'émergence d'un nouveau modèle de gestion de la propriété intellectuelle qui tend à s'étendre à de nouveaux domaines comme la publication de documents.

Les débuts libres du logiciel

Au commencement de l'informatique, les grands constructeurs comme IBM ou DEC aidaient leurs clients à prendre en main les logiciels en leur fournissant les codes sources et en animant des clubs utilisateurs pour favoriser l'enrichissement et l'échange des logiciels entre clients. Certains de ces clubs, comme Decus (Digital Equipment Computer Users Society), sont considérés comme les précurseurs des communautés de logiciels libres actuelles. Du milieu des années 1970 au milieu des années 1980, la vision de plus en plus restrictive de la propriété intellectuelle sur le logiciel a fini par aboutir au mode des licences

Succès

La vision novatrice et très ouverte de la propriété des logiciels proposée par la FSF a permis la mise en place de nombreux groupes d'intérêts sur les « souches » de logiciels libres. Ces groupes sont appelés des « communautés », conformément à l'esprit qui les anime. Les plus connues de nos jours sont GNU-Linux, Apache, Mozilla (Firefox, Thunderbird), Document Fondation (LibreOffice).

dites « propriétaires », actuellement majoritaires, avec diffusion des seuls exécutables, boîtes fermées intouchables par les utilisateurs. La micro-informatique a été un moteur puissant pour le développement de ce modèle, qui n'était pas une première sur le logiciel.

Le logiciel libre, un rêve *flower power* ?

Au milieu des années 1980, certains informaticiens, se rendant compte du changement de modèle, s'y sont opposés. Un en particulier, Richard Stallman, a posé les fondements du logiciel libre avec la création de la Free Software Foundation. Partant du principe que le logiciel allait être au centre du fonctionnement du monde à venir, ce qui se confirme, il voulut établir les règles qui permettraient de garantir que les utilisateurs resteraient maîtres de ce qui devait rester un outil, et resterait donc « libre ». Ces règles prennent différentes formes mais, au fond, établissent la liberté d'exécuter le programme pour tous les usages, d'étudier le fonctionnement du programme et de l'adapter à ses besoins, de redistribuer des copies du programme, et enfin d'améliorer le programme et de distribuer ces améliorations au public pour en faire profiter toute la communauté. Cela implique bien sûr que le code source soit librement accessible.

Le logiciel sera au centre du fonctionnement du monde à venir

REPÈRES

En 1984, Richard Stallman lance le développement de système d'exploitation libre baptisé GNU (Gnu's Not Unix). Les utilisateurs doivent accepter les conditions d'utilisation définies dans la « GNU General Public License » (GNU GPL), contrat d'adhésion qui constitue aujourd'hui le modèle de la plupart des logiciels libres. Le *copyright* étant la base de la protection des logiciels « propriétaires », Richard Stallman propose le terme de *copyleft* pour définir les droits des utilisateurs.



© MICHA L. RIESER

Le gnu (en anglais *gnu*), emblème de la communauté GNU/Linux issu d'une définition récurrente : *What is GNU ? Gnu is Not Unix.*

caractéristiques (effet contaminant ou non, *multilicensing*, droit applicable, garanties juridiques) varient en fonction de leur auteur, mais ces licences sont opposables aux utilisateurs et commencent à être reconnues, y compris devant les tribunaux.

Et pourtant, pas si naïf

Mais, contrairement à une idée reçue, le recours à des logiciels libres ne signifie en rien que les utilisateurs n'aient aucune obligation à respecter. Un logiciel libre n'est pas libre de droit, puisqu'il a un créateur. Les initiateurs du logiciel libre, réalistes, se sont insérés dans le monde du droit en formulant dans des licences les droits et obligations qui s'appliquaient. Plusieurs grands types de licences ont été définis, dont les principaux sont la GNU General Public License (GPL), la Berkeley Software Distribution (BSD) et la licence Apache. Il en existe aussi une en droit français, la licence CEA CNRS INRIA Logiciel libre (CECILL). Les

Copyleft

Ce terme désigne un ensemble d'obligations contractuelles reposant sur un principe simple : si je diffuse librement quelque chose, j'interdis aux autres de le privatiser et de l'utiliser dans leur seul intérêt. Cela implique donc que, si je modifie un logiciel, je dois mettre à disposition toutes les modifications que j'ai effectuées. La contribution à l'effort collectif devient une obligation et permet de maintenir la dynamique de développement.

Jurisprudence

Un arrêt de la cour d'appel de Paris du 16 septembre 2009 a condamné une société informatique aux motifs qu'elle n'avait pas fourni à son client (l'AFPA) les sources d'un logiciel libre et avait supprimé le texte de la licence GNU-GPL. Suite à un appel d'offres important (plusieurs millions d'euros), l'AFPA a retenu la société EDU4. Lors de la recette de la phase 1 du marché, l'AFPA découvre que le logiciel VNC est un logiciel libre qui a été modifié par la société EDU4 et elle demande les sources à plusieurs reprises au prestataire, y compris par l'intermédiaire de la Free Software Foundation France. Ce dernier refuse d'accéder à sa demande et, lorsqu'il les fournit, ce ne sont pas celles qui correspondaient à la version livrée en 2001. Cette décision met en exergue l'inexécution des obligations contractuelles justifiant la résolution du contrat aux torts exclusifs du prestataire informatique.

Contrat d'adhésion

Qui télécharge une licence de logiciel libre se trouve dans le cas d'un contrat d'adhésion, c'est-à-dire dans la même situation qu'en cas d'achat d'un logiciel propriétaire (par exemple, la *shrink wrap license*, où l'acheteur du produit est lié par la licence du simple fait d'avoir déchiré l'emballage). Imposées par l'auteur, les clauses de la licence sont non négociables. *In fine*, soit le licencié accepte la licence et peut faire ce qui y est mentionné, soit il ne peut pas bénéficier de toutes les libertés inhérentes au logiciel libre (modification et distribution).

Toute modification ou adaptation peut avoir des implications fortes. C'est un des points importants, et pourtant souvent négligés, du logiciel libre. Il convient de connaître les obligations associées à un logiciel libre, en particulier en cas d'utilisation dans un système d'information professionnel. Les décideurs publics ou privés et les sociétés de services (SSII ou SSLL) qui recourent à des logiciels

**Un logiciel libre
n'est pas libre
de droit**

- libres doivent donc porter une attention particulière au choix des licences et à ses conséquences juridiques. Le simple utilisateur en revanche, qui ne modifie en rien le logiciel, est toujours couvert par une entière liberté d'usage.

Un autre modèle de propriété intellectuelle

Ce qui pouvait être considéré comme un doux rêve de *baba cool* est en fait un véritable modèle alternatif de gestion de propriété intellectuelle pensé avec une réelle efficacité. Comme tout modèle de propriété intellectuelle, il tend à s'auto-entretenir : le logiciel libre le plus diffusé l'est selon le mode dit *copyleft* de la licence GNU-GPL, qui protège sa nature de logiciel libre.

Les utilisateurs d'abord

Ce modèle est orienté par les besoins des utilisateurs. Une communauté n'a pas intérêt à développer une fonction qui n'est utile qu'à très peu d'utilisateurs au sein d'un logiciel libre. Alors que le changement de version régulier est une source de revenu importante pour le logiciel propriétaire, que l'utilisateur en ait ou non besoin, la stabilité est une qualité pour un logiciel libre. La règle est donc la mise en commun des besoins et la priorisation des évolutions. Autre avantage du modèle : il permet de résister aux tentatives de détournement. Dans certaines communautés libres, des acteurs du logiciel propriétaire sont très actifs. L'intérêt propre de leur société peut les amener à vouloir orienter les développements en s'éloignant de l'intérêt de la communauté. Le modèle libre permet alors à une partie de celle-ci de faire ce que l'on nomme un *fork*, c'est-à-dire repartir du code source du moment dans une autre direction de développement.

Privilégier le travail local d'amélioration plutôt que la rente globale

Un *fork* emblématique

La suite bureautique OpenOffice a été un exemple majeur de la capacité d'autodétermination des membres d'une communauté de développement libre. Face aux orientations d'Oracle, jugées contestables par une partie de la communauté, un *fork* a donné lieu à la création de l'alternative LibreOffice, souche qui, un an après, prend clairement le pas sur l'original.

Émulation et créativité

Autre mérite du modèle : il crée l'émulation nécessaire pour stimuler la créativité. Que ce soit par un *fork* ou en s'appuyant sur l'ensemble des logiciels libres existants, ceux qui sont sûrs d'avoir une bonne idée peuvent toujours se lancer avec un faible investissement et réunir une communauté autour de cette idée.

C'est ainsi que de nombreuses souches se créent en permanence. Seules survivront celles qui sont suffisamment pertinentes pour être portées par un grand nombre de développeurs et d'utilisateurs.

Une activité économique locale

Enfin, ce modèle privilégie le travail local d'amélioration plutôt que la rente globale. Pour un usage au sein d'un système d'information professionnel, le logiciel libre n'est pas gratuit ; il faut en effet l'intégrer et s'assurer d'avoir un support de qualité.

Pour tout cela, il est souvent fait appel à des sociétés de services dont certaines se sont spécialisées en sociétés de services logiciel libre (SSLL). Il y a là injection d'argent pour entretenir la dynamique. Le client paie un fournisseur proche de lui pour une insertion au mieux dans son environnement, et contribue aussi à l'amélioration de la souche, au lieu de payer une licence à un fournisseur bien loin de lui.

Libre ou pas libre

Le modèle libre tend aujourd'hui à s'étendre à d'autres domaines comme les publications de documents ou d'œuvres (licences *Creatives Commons*), ou même le matériel (OSHW, OpenSource Hardware). Il s'appuie sur l'intelligence collective et revendique la prise en compte de l'intérêt des utilisateurs. C'est clairement un modèle de coopération, et il sait rappeler à l'ordre ceux qui oublient les règles du jeu collectif.

Le modèle libre et le modèle propriétaire coexistent, et il est important de prendre en compte toutes leurs possibilités et limitations pour juger de leur efficacité technique et économique. Toutefois, pour les logiciels qui couvrent des fonctions très courantes voire standardisées, y compris dans les centres informatiques, le logiciel libre pourrait s'imposer de fait. À moins que le brevet logiciel, tant contesté, n'arrive à le museler. ■