

FLASH

INFORMATIQUE

p/a EPFL – SERVICE INFORMATIQUE CENTRAL – CP 121 – CH 1015 LAUSANNE – tél. +41 21 693 22 11 – web: <http://sic.epfl.ch>



Spécial ÉTÉ 2001
Logiciels libres



EDITORIAL

ANNE.POSSOZ@epfl.ch, SERVICE INFORMATIQUE CENTRAL



L'émergence des logiciels libres constitue un événement passionnant dans notre société, tant par les perspectives qu'ils offrent que par les questions qu'ils posent. Ils nous obligent à faire preuve d'inventivité pour modifier aussi bien nos modes de fonctionnements que nos lois, afin de tenir compte des nouvelles caractéristiques de communication et de copies sous forme numérique.

L'innovation la plus marquante dans ce domaine est sans doute aujourd'hui la licence GPL (GNU General Public License) [1] qui utilise de façon astucieuse les mécanismes de licence des logiciels pour assurer qu'un logiciel libre puisse le rester.

L'enjeu de notre attitude face à ce phénomène est tel qu'il nous a paru opportun d'y consacrer le numéro spécial de cette année.

Rappelons que ce qui caractérise un logiciel libre est la liberté d'exécuter, de copier, de distribuer, d'étudier, de modifier et d'améliorer le logiciel. La licence GPL prend acte lors de la modification et de la distribution d'un logiciel publié sous cette licence [2].

Si l'arrivée de Linux a cristallisé la possibilité d'avoir un système d'exploitation entièrement libre, les logiciels libres avaient déjà une histoire, depuis la fondation du projet GNU [3] par Richard Stallman en 1984. C'est pourquoi, quand on parle du système d'exploitation, on parle de GNU/Linux, Linux n'étant lui-même que le noyau de ce système. La richesse des logiciels libres applicatifs disponibles pour GNU/Linux ne cesse de croître. La liste est bien trop longue pour être énumérée ici mais une bonne source de référence est sans doute celle du projet GNU [4].

Nous avons rassemblé ici certains articles originaux et d'autres déjà publiés dans certaines revues ou sur la toile, toujours en accord avec leurs auteurs, afin de constituer une référence solide sur le thème.

Les réflexions sur la propriété intellectuelle (pages 11-16), le droit d'auteur et le copyright (pages 63-72), y compris leur internationalisation (pages 57-58) permettent de bien saisir la démarche. Les aspects juridiques évoqués dans ces textes devront d'ailleurs être pris en compte assez rapidement au sein de l'EPFL. Si le mouvement du logiciel libre se renforce de façon spectaculaire aux États-Unis d'Amérique, l'Europe n'est pas en reste (pages 23-26) et son déploiement concerne plus que jamais le reste de la planète. Les logiciels libres sont aujourd'hui promus dans

l'enseignement en France (pages 27-30) et leur utilisation dans les entreprises est devenue une réalité tangible (pages 3-10 et 17-22). Des licences proches de la GPL sont parfois l'unique chance de sauver le logiciel de pointe que les propriétaires abandonnent (pages 59-62). Dénrée périssable, tel est le titre de la nouvelle primée cette année (pages 42-44) et qui illustre malicieusement les textes de ce numéro. Mais revenons sur terre pour constater que des chercheurs se penchent sur des modèles de viabilité économique du mode de développement de ces logiciels (pages 31-40). Des applications scientifiques basées entièrement sur des logiciels libres se développent (pages 45-50 et 51-56) au cœur même de notre institution.

Lors de la rencontre du 12 juin 2001 sur les logiciels libres à l'EPFL, Roberto Di Cosmo nous a brillamment démontré les principales raisons de passer du logiciel propriétaire au logiciel libre : sécurité, robustesse, autonomie, pérennité, partage des connaissances, partage des coûts de développement, protocoles et formats de données publiés, tout en nous mettant en garde sur le danger des brevets de logiciels [5]. Une remarque faite lors de cette rencontre est ici incluse et pourrait nous servir de conclusion (page 73).

Pour donner suite à ces lectures, rendez-vous sur le site Logiciels libres de notre Ecole, elle.epfl.ch.

RÉFÉRENCES

- [1] <http://sic.epfl.ch/publications/FI00/fi-10-00/10-00-page10.html>
<http://www.gnu.org/licenses/licenses.html#GPL>
- [2] Extrait de la licence GNU GPL, <http://www.gnu.org/licenses/gpl.html>: «You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.»
- [3] <http://www.gnu.org/>
- [4] <http://www.gnu.org/directory/>
- [5] Un exposé proche de celui qu'il nous a donné: <http://www.pps.jussieu.fr/~dicosmo/TALKS/Orsay/>. L'exposé lui-même sera accessible à partir du site <http://elle.epfl.ch> ■

L'ENTREPRISE DOIT-ELLE ADOPTER LES LOGICIELS LIBRES?



GILBERT ROBERT, PRÉSIDENT@linux-gull.ch, FRÉDÉRIC SCHÜTZ, SCHUTZ@MATHGEN.CH,
GULL – GROUPE DES UTILISATEURS LINUX DU LÉMAN [1]



Plus que jamais, le phénomène du logiciel libre fait parler de lui, médiatisé par le nombre croissant d'articles à son sujet dans différents journaux et par les récentes prises de position en sa faveur de la part de grandes entreprises telles que IBM, Sun ou Hewlett-Packard.

Le GULL, Groupe des Utilisateurs Linux (et des Logiciels Libres) du Léman, a participé pour la deuxième fois cette année à l'exposition Computer à Lausanne et peut déjà tirer une première conclusion: le logiciel libre et son navire amiral Linux interpellent de plus en plus de personnes, aussi bien les utilisateurs particuliers que les informaticiens des multinationales. Les besoins ne sont certes pas identiques mais les mêmes questions reviennent. Qu'est-ce que Linux? Qu'est-ce que cela va apporter de plus à mon entreprise? Est-ce compatible avec Windows? Quelles sont les applications qui tournent? On m'a dit que ce n'était ni sûr ni fiable? Est-ce que le support existe?

Le but de cet article est d'expliquer plus en détails ce que sont les logiciels libres et leurs avantages pour les utilisateurs, et de faire un survol de leur position dans les entreprises, en examinant en particulier les obstacles qui s'opposent encore à leur adoption.

QU'EST-CE QU'UN LOGICIEL LIBRE?

Contrairement à ce que peut laisser supposer l'ambiguïté de la langue anglaise, la caractéristique principale du *free software* n'est pas d'être gratuit, mais bien d'être libre. Pour illustrer cette différence, on peut comparer le code source (*les sources*) d'un programme à une partition de musique alors que le programme lui-même sera l'équivalent de son exécution par un orchestre. Dans le cas d'un logiciel libre, vous disposez de la partition et vous pouvez la rejouer, de la même manière ou avec un autre instrument ou une autre orchestration, alors que dans celui d'un logiciel *propriétaire*, vous n'entendez que la musique et ne pouvez pas corriger

d'éventuelles fautes d'accord ou adapter le tout à votre goût. Un concert peut tout à fait être gratuit, même si le compositeur refuse de diffuser la partition de son œuvre (ce serait le cas d'un logiciel *propriétaire* distribué gratuitement, mais sans son code source, tel que Microsoft Internet Explorer), alors qu'à l'inverse, s'il a accepté de diffuser sa partition, ceci n'impose nullement la gratuité du concert (cas d'un logiciel libre, pouvant être vendu, mais accompagné de son code source).

Contrairement à une opinion répandue, le logiciel libre n'est pas simplement placé dans le domaine public par son auteur, qui abandonnerait ainsi tous ses droits, mais il est soumis à une licence qui détermine les droits et devoirs de ceux qui l'uti-

lisent. La licence la plus connue et la plus utilisée est la *GNU General Public License* (GNU GPL), définie par Richard Stallman, fondateur de la *Free Software Foundation* (FSF). Elle place l'utilisation de logiciel dans un cadre juridique, humoristiquement appelé le *copyleft* en opposition au *copyright* duquel se réclament les licences habituelles. En diffusant son logiciel sous cette licence, un auteur garantit à chaque utilisateur **quatre libertés**, qui définissent un logiciel comme étant libre:

- la liberté d'exécuter le programme, quel qu'en soit l'usage;
- la liberté d'étudier le fonctionnement du programme et de l'adapter à vos besoins, ce qui nécessite **l'accès aux codes sources**;
- la liberté de redistribuer des copies;
- la liberté d'améliorer le programme et de publier vos améliorations.

L'utilisateur d'un logiciel a ainsi autant de droits que son auteur, y compris celui de revendre le logiciel et les modifications qu'il y a apportées. La seule contrainte que lui impose la GPL est qu'il ne peut pas priver d'autres utilisateurs de ces libertés, c'est-à-dire que s'il diffuse des versions modifiées du logiciel, il est également obligé de diffuser le code source de ses modifications. Mais cette contrainte ne s'applique qu'aux modifications du logiciel lui-même, et pas à d'autres programmes qui inter-agiraient avec lui, contrairement à ce que Steve Ballmer, le CEO de Microsoft, décrivait récemment en déclarant que «*Linux est comme un cancer (sic) qui s'attache à tout ce qu'il touche*». Ainsi, le fait que GNU/Linux soit un système d'exploitation diffusé sous licence GPL ne signifie pas que

le code source de tout programme qui fonctionne sous Linux doit obligatoirement être rendu public !

Il existe d'autres types de licences pour logiciels libres, telle que la licence BSD, qui au contraire de la GPL, autorise un développeur à redistribuer une version modifiée d'un programme sans devoir rendre publiques les modifications qu'il a apportées. Les éditeurs peuvent ainsi intégrer des parties de logiciels libres dans des produits qui ne le sont pas, ce qui a par exemple permis à Microsoft de réutiliser dans ses programmes, Windows 2000 entre autres, des morceaux de codes provenant de logiciels libres tels que le système d'exploitation FreeBSD, et ceci de façon tout à fait légale.

QUELS SONT LES AVANTAGES DES LOGICIELS LIBRES DANS LE CADRE D'UNE ENTREPRISE?

Les logiciels libres possèdent des atouts techniques non négligeables à faire valoir dans l'entreprise, en particulier grâce à la disponibilité des sources. Parmi les plus souvent cités, on trouve:

- **l'utilisation de standards ouverts**, et le respect de ces standards, permettant une compatibilité entre différents produits;
- **une sécurité accrue**;
- **l'adaptation à de nombreuses plates-formes** (PC, Mac, Sparc, Alpha, mainframes IBM S/390 ou même systèmes embarqués).

Contrairement à la plupart des solutions non-libres, l'adoption d'un système GNU/Linux, par exemple, ne nécessite pas une métamorphose radicale du système informatique de l'entreprise, et on peut envisager une migration incrémentale service par service. Ainsi, un serveur de fichiers qui utilise le logiciel libre Samba [12] peut remplacer avantageusement un serveur de domaine NT, et n'empêche pas de garder ses postes clients sous Windows. Une ancienne machine, par exemple de type Pentium 133, peut très bien être recyclée à l'aide de logiciels libres comme firewall ou comme serveur de courrier électronique. A l'inverse, Alan Cox, un des principaux développeurs

du noyau de Linux, soulignait récemment: «*Choisissez un serveur Microsoft, et vous devrez choisir un client Microsoft. Choisissez Microsoft Project et vous devrez utiliser un système d'exploitation Microsoft, et pourriez bien avoir à utiliser de tels systèmes sur la moitié de vos ordinateurs.*».



A l'heure où la plupart des postes de travail sont connectés à Internet, un des arguments-clé qui plaident en faveur des logiciels libres est la sécurité informatique. Ce problème majeur, mais peu visible la plupart du temps, est souvent sous-estimé ou placé au second plan des préoccupations. Ceci est d'autant plus grave que la majorité des gens pense encore que la meilleure sécurité est fournie par les logiciels propriétaires, car les éventuels pirates ne peuvent pas utiliser les sources pour détecter les faiblesses. Cette vision de **sécurité par l'obscurité** ne tient pas quand on compare le nombre de problèmes de sécurité qui affectent des systèmes libres ou non-libres. Ces dernières années, des problèmes de sécurité détectés par des experts indépendants dans des systèmes aussi stratégiques que les cartes bancaires françaises ou le chiffrement des communications des téléphones portables GSM, et dont les spécifications étaient pourtant secrètes, ont montré que ce principe de la *boîte noire* servait principalement à cacher les faiblesses du système à ses utilisateurs légitimes, qui n'ont aucun moyen de vérifier par eux-mêmes la sécurité des systèmes qu'ils utilisent et sont obligés de croire sur parole les promesses des concepteurs. Et ces derniers, s'ils découvrent un défaut dans un de leurs produits, peuvent être tentés de le cacher plutôt que de ternir leur image de marque. Cela peut sembler paradoxal, mais même dans le domaine hautement sensible de la cryptographie, aucun système n'est considéré comme sûr s'il n'a pas été proposé publiquement à l'étude de la communauté internationale pendant

plusieurs années sans qu'une faille ne soit trouvée, et aucun expert n'acceptera de cautionner un système non-public.

Un logiciel libre ne sera pas intrinsèquement plus sûr qu'un autre, même si la disponibilité du code source permet d'augmenter le nombre de personnes aptes à découvrir des défauts et à les corriger. Mais si un défaut est trouvé et une correction proposée, celle-ci peut être diffusée rapidement à tous les utilisateurs, qui ne dépendent pas du bon vouloir de l'éditeur original. Après leur sortie sur le marché, les bugs d'un logiciel libre seront donc plus rapidement corrigés, ce qui donnera dans les faits des logiciels plus sûrs. Un exemple extrêmement parlant est celui d'Interbase, une base de données distribuée par Borland. Pendant plusieurs années, elle a été vendue munie involontairement d'une *backdoor*, un défaut caché qui permettait à un pirate d'accéder facilement à toutes les données contenues dans le système sans avoir le mot de passe. Quand Borland a ouvert le code source de son programme, le problème a été découvert et corrigé très rapidement, alors qu'il aurait pu rester caché des utilisateurs (mais pas forcément des pirates) pendant encore longtemps.

Mais pour une entreprise, hormis ces qualités techniques, d'autres facteurs importants peuvent rentrer en compte:

- **l'indépendance par rapport à un éditeur de logiciel et de sa politique**, puisque l'on dispose des codes sources, des spécifications et que les standards sont suivis à la lettre (dynamique des outils garantie, fiabilité et sécurité plus grandes);
- le fait de n'avoir **plus de licence compliquée à gérer et à payer**, de ne plus risquer d'être hors-la-loi. Les mises à jour peuvent ainsi être planifiées sans contraintes et uniquement si cela est nécessaire.

Concernant l'indépendance, les exemples d'entreprises dont le fournisseur informatique a fait faillite ne manquent pas. Sur le plan romand, plusieurs PME se sont ainsi retrouvées avec des logiciels (de comptabilité et gestion par exemple) figés qu'il était impossible de modifier. En cas de problème ou de besoin nouveau,

la seule solution est alors de le remplacer entièrement par un nouveau système, mais ce double investissement est hors de portée de la plupart des entreprises, et rien n'empêche le même scénario de se répéter. A l'inverse, les utilisateurs qui ont décidé d'utiliser le gestionnaire de fichiers Nautilus n'ont pas ce problème, car bien que la société Eazel, son créateur, ait été emportée dans la tourmente des faillites des sociétés **.com**, le fait que le logiciel soit libre garantit qu'il pourra quand même évoluer.

Au sujet des licences complexes et difficiles à suivre, un membre de l'AFUL, l'Association Francophone des Utilisateurs de Linux et des Logiciels Libres [2], révélait dernièrement que la Business Software Alliance (BSA) – <http://www.bsa.org> et Microsoft avaient redressé plusieurs établissements scolaires français pour non conformité, ce qui a entraîné des prises de position de recteurs, de conseils généraux et régionaux en faveur des logiciels libres et de GNU/Linux. De nombreux établissements utilisent désormais la suite bureautique StarOffice – <http://www.staroffice.org> et n'achètent plus les mises à jour de Microsoft. Dans l'académie d'Amiens par exemple, tous les serveurs de type *proxy* (qui stockent les pages Web pour en accélérer la rediffusion dans un réseau interne) sont passés sous GNU/Linux en 2001 et des formations à l'installation de stations de travail et de serveurs sécurisés sous GNU/Linux ont été organisées par des instituts de formation des maîtres.

Maintenant si l'on supprime cette épine des licences logicielles, **l'entreprise peut déplacer son budget sur les services et la personnalisation et consacrer plus de moyens aux conseils, à la formation, et à l'assistance technique.** Si l'installation peut être plus chère, on peut considérer ensuite que l'intégration, la personnalisation, la maintenance, la sécurisation, l'évolution et l'archivage seront beaucoup plus aisés et moins chers à mettre en œuvre. Les entreprises habituées aux problèmes récurrents de maintenance informatique, peuvent à nouveau établir une relation de confiance et de proximité avec les sociétés qui se développent sur la base de ce nouveau modèle économique des logiciels li-

bres. Ces nouvelles sociétés doivent se rapprocher des entreprises pour offrir un service autour de la mise en place rapide et à moindres frais de serveurs (tels que serveur de fichier et d'impression, serveur Web et de courrier électronique, ou encore de *firewall*), autour de développements rapides d'applications personnalisées grâce à **la boîte à outils** des logiciels libres: Perl, Apache, Mysql, PostgreSQL, PHP, Zope, Python, Postfix,...

C'EST PUISSANT, FIABLE, OUVERT ET ALORS?

Pour une entreprise, l'argument du *libre* n'est *a priori* pas très important, comparé à l'avantage d'avoir un interlocuteur unique avec un nom connu, la garantie d'un support technique, d'une *hotline* et d'un prix. Les considérations philosophiques et les *guerres de religion informatique* ne font pas partie des préoccupations des décideurs, et les pressions hiérarchiques ou commerciales ont un poids que l'on sous-estime souvent. Pourquoi chercher d'autres solutions dont on ne connaît pas grand chose, lorsqu'un paquet prêt à l'emploi vous est proposé? Pourquoi prendre des risques quand on peut se couvrir simplement en choisissant le leader mondial dans le domaine?

Quels choix le marché offre-t-il? Tout d'abord, du côté des serveurs, il y a la solution Microsoft Windows qui représente environ 40% du marché. Il y a ensuite les Unix propriétaires, comme ceux produits par Sun ou IBM, qui sont certainement très pointus, adaptés aux gros systèmes critiques, fiables, mais trop chers, et les compétences sont rares. Aujourd'hui, il y a GNU/Linux, et d'autres Unix libres, qui ne sont pas encore très connus, et que l'on peut se procurer gratuitement, ou en payant un faible prix si l'on veut acheter un CD original et un manuel imprimé (moins de 100 CHF pour une distribution Linux complète), ce qui ne donne pas une impression de sérieux quand on les compare aux budgets informatiques habituels. Du côté des postes de travail, le choix est encore plus restreint, puisque plus de 85% du marché est détenu par Microsoft Windows, contre environ 6% pour

MacOS. D'après vous dans quelle voie ce décideur va-t-il engager l'entreprise?

Cependant, on peut actuellement entendre un autre son de cloche. Les mentalités et les responsables informatiques changent et les esprits s'ouvrent. A l'occasion de Computer, les membres du GULL ont pu ainsi entendre de nombreuses réactions positives sur l'adoption d'un système informatique basé sur les logiciels libres.

Outre un certain *ras-le-bol des plantées à répétition et des mises à jour coûteuses*, on a pu relever des arguments relatifs aux coûts indirects (partie du TCO) que les décideurs oublient trop souvent de prendre en compte. Ces coûts indirects engendrés par une maintenance plus importante que prévue, les changements de stratégie des entreprises éditrices de logiciels propriétaires, les adaptations aux problèmes de l'entreprise qui n'ont pas été prévues ou les problèmes liés à la sécurité informatique (virus, intrusions).

En ce qui concerne la maintenance, une étude menée à l'école d'ingénieurs de Marseille montre que le temps nécessaire à la maintenance logicielle de serveurs Windows est quasiment le double de celui de serveurs de type Unix. Dans les faits, la maintenance basique (assistance utilisateur, et maintenance d'un parc de machines sous Microsoft Windows) correspond à plus de 60% du travail d'un ingénieur système, au lieu de se consacrer à la maintenance des serveurs, à la sécurité du site, à la veille technologique, la rédaction de documents et à la réflexion stratégique à long terme. En Suisse une étude de Swepix [17] a d'ailleurs montré que parmi tous les serveurs Web testés (ceux de banques, assurances, administrations et entreprises diverses), les serveurs sous Windows IIS tombaient en moyenne 2 fois plus souvent en panne qu'un serveur Apache [10].

C'est sur la réduction des coûts engendrée par l'adoption des logiciels libres que l'on appuie le plus souvent dans les comparaisons entre les différentes solutions. Les entreprises sont bien entendu très sensibles à cette argumentation, mais on peut se demander si finalement les avantages qui pourraient devenir les plus significatifs ne se situent pas ailleurs et notam-

ment dans la **pérennité des données** et dans la **sécurité**.

La **pérennité des données et des protocoles** est importante pour garantir l'interopérabilité des applications et des documents. On est toujours sensible au fait que l'on ne puisse pas lire le document de son voisin qui a une version différente du même logiciel. On peut se poser la question *est-ce que mes données qui sont sauvegardées dans un format propriétaire seront toujours lisibles dans 10 ans? ou est-ce que ce logiciel sera toujours exploitable dans 2 ou 3 ans?* A ces questions la réponse est incertaine puisque l'on n'a aucune maîtrise de l'application métier que l'on utilise et des documents générés. Ainsi, avec l'accélération de la société de l'information, on peut mettre en péril à moyen ou à court terme l'investissement informatique de l'entreprise. Il faut donc prendre en compte dans sa stratégie le facteur important que constituent les formats des données. Si ces formats sont basés sur des standards dont on a les sources alors on pourra toujours les réutiliser, les transcrire pour les exploiter à nouveau et cela même après 20 ans. A ce sujet, on fait beaucoup mention actuellement de XML. C'est certainement une révolution dans la représentation de l'information et il est fort à parier que dans l'avenir beaucoup d'outils seront basés sur ce standard. Encore faut-il, pour des raisons de compatibilité, que le standard XML soit le même pour tout le monde!

Au niveau de la sécurité informatique, les grosses entreprises n'ont aucun mal à globaliser les problèmes liés à l'accès au réseau Internet. Ce qui n'est pas le cas des petites et moyennes entreprises (PME), pour qui sécurité rime souvent avec anti-virus. Or dans les deux cas il faut penser, comme nous le mentionnons dans le chapitre précédent, que la sécurité passe par la maîtrise de son environnement informatique et donc par la disponibilité des codes sources et de la documentation (volumineuse dans le cas des logiciels libres). La sécurité informatique n'en est qu'à ses débuts, à l'heure où l'on parle de e-gouvernement, de e-voting ou de e-commerce, il faudra prendre conscience des dangers que constituent les intrusions sur les systèmes informati-

ques internes et les dégâts considérables que peuvent causer les virus.

Après avoir mis en avant les principaux atouts des logiciels libres, il n'est pas de meilleure méthode pour se convaincre que de regarder les expériences que son voisin un peu plus entreprenant a réalisées. A Genève, le CERN a choisi de diminuer l'hétérogénéité de son parc informatique et de maintenir un ensemble de 6000 machines GNU/Linux, 3500 Windows, 3000 Unix et 1200 imprimantes. Dans ce contexte scientifique, les logiciels libres sont un choix fréquent pour les utilisateurs qui ont de gros besoins en temps de calcul, en particulier parce qu'il est possible de paramétrer le système pour que le maximum de ressources soit disponible pour les calculs (par exemple, une interface graphique est inutile sur un tel système et ne fait que ralentir les calculs). On peut aussi citer l'Observatoire de Genève [14], qui a installé une ferme de 64 machines utilisant GNU/Linux et travaillant en commun pour effectuer des simulations astronomiques, Shell, qui utilise une ferme de 1024 serveurs IBM sous GNU/Linux pour la recherche pétrolière, et à l'extrême, le moteur de recherche Google – <http://www.google.com> – qui doit sa réussite à l'emploi d'une ferme de 8000 serveurs utilisant ce même système! Lors de la journée sur les logiciels libres, organisée par l'observatoire technologique de l'Etat de Genève et le GULL [19], Edouard Soriano, directeur de la société DAPSYS, a expliqué comment, grâce au choix de Linux dans un environnement critique tel que l'imagerie médicale à la clinique des Grangettes, il a pu s'extraire des solutions propriétaires coûteuses et assurer à la fois un avenir économique plus serein et un accès aux données garanti à long terme.

Dans le même registre, le musée du Louvre a décidé de développer une solution de numérisation de l'ensemble des œuvres qui sera basée sur les logiciels libres, ceci afin de rendre plus pérenne l'investissement et d'éviter des changements de stratégie des solutions propriétaires de gestion de base de données qui coûtaient une fortune. Cette solution utilisant les logiciels libres permettra de stocker les images sur un serveur de fichiers

Linux de 2 To et de les indexer dans une base documentaire en quelques semaines. La solution conçue par HP pourra être diffusée gratuitement à tous les autres musées du monde.

On pourrait encore citer beaucoup de grandes entreprises dans la liste de celles qui ont fait le choix de GNU/Linux, telles IKEA, le Figaro, Mercedes, Boeing ou encore l'Oréal, mais on y trouve aussi des PME qui ont choisi cette voie pour gérer l'accès sécurisé à Internet pour l'Intranet, le partage des fichiers, des imprimantes ou encore pour gérer l'information à l'aide d'un serveur Web et un serveur de courrier électronique.

Longtemps les logiciels libres ont dû faire face à une critique fondée sur le manque de support et de maintenance professionnels. Il est en effet impensable qu'une entreprise envisage de faire un pas en direction des solutions libres si aucune garantie de support informatique ne peut être offerte. Devant ce manque, plusieurs entreprises se sont créées, tout d'abord du côté français avec entre autres, des sociétés telles Alcôve [7], Easter-Eggs [8] ou Aurora – <http://www.aurora-linux.com/>, et plus récemment et plus près de chez nous, en Suisse romande, avec des sociétés telles que Goelaan [5] ou ProLibre [6].

Comme on peut le constater, les arguments ne manquent pas pour adopter les logiciels libres dans une entreprise. La route toujours plus large des logiciels libres, pavée de qualités techniques indéniables et d'offres professionnelles grandissantes, est cependant entravée par des obstacles d'un tout autre registre.

LES OBSTACLES

Il existe évidemment des obstacles à l'adoption des logiciels libres dans les entreprises et les administrations. Un des plus connus est sans nul doute la **méconnaissance du phénomène** des logiciels libres, soigneusement entretenue par différents courants dont le premier est sans surprise Microsoft, qui a récemment publié un document intitulé *Linux in Retail & Hospitality* [20], dont voici quelques courts extraits:

- *Less Secure: Open source means that anyone can get a copy of the source*

code. Developers can find security weaknesses very easily with Linux. The same is not true with Microsoft Windows. Les trous de sécurité sous Linux sont effectivement découverts par les experts avant les *crackers*, et les mises à jour publiées très rapidement et non l'inverse.

- *Support / Maintenance Costs: Support and maintenance for Linux is not free. Most Linux distributors make their money by selling their services. Support options vary by vendor and can get quite expensive for the enterprise. You will have to pay for support when you need it.* Même s'il est tout à fait exact que les sociétés de service autour de Linux facturent le support et les services, l'article semble sous-entendre que ces mêmes services sont gratuits pour la plate-forme Windows.
- Citons pour finir un paragraphe qui se passe de commentaires quand on compare la structure ouverte et libre du développement des logiciels libres avec les développements centralisés conduits par Microsoft: *Lack Of Formal Development Schedule, Research, and Standards: With Linux, no formal development schedule or set of standards exists. There are thousands of developers contributing to it from all over the world, with no accountability to the retail industry. Linus Torvalds makes the final decision about what gets included in the latest Linux release, and he has no accountability to the retail industry. There is no formal research and development process with Linux. Microsoft plans to spend over \$4 billion in R&D in 2001 and listens to the retail industry.*

Il s'agit clairement ici de désinformation, et on quitte le domaine technique pour entrer dans la propagande. La méthode utilisée consiste à créer le doute et la peur de choisir des solutions incompatibles avec le *standard*, n'hésitant pas pour cela à inverser les rôles. Les logiciels libres étant réputés justement pour leur compatibilité avec les standards. En conjonction, Microsoft utilise aussi une technique qu'ils ont eux-mêmes appelée *Embrace and Extend* afin de fidéliser leurs clients contre leur volonté. Cette

technique consiste à utiliser un format de données ou un protocole d'échanges standards et ouvert qui a fait ses preuves, mais à le modifier légèrement pour qu'il soit incompatible avec d'autres logiciels. C'est le cas par exemple du serveur de courrier électronique MS Exchange, ou du protocole de sécurité Kerberos.

L'autre contre-courant, plus sérieux et plus subtil, se situe sur le **terrain juridique**. Devant leur impuissance à contrôler et à maîtriser ce mouvement, les grands groupes tentent d'utiliser l'arme commerciale que constitue les brevets appliqués aux logiciels. En théorie, il n'est pas possible actuellement en Europe de breveter des logiciels, ceux-ci étant protégés de la copie non-autorisée par les lois sur le droit d'auteur. Au contraire, les Etats-Unis ont étendu le domaine de la brevetabilité aux biens immatériels tels que les logiciels et les méthodes d'affaires, et des milliers de brevets ont été déposés, la plupart pour des méthodes triviales et très générales. Contrairement au droit d'auteur, qui ne protège que le logiciel lui-même, ces brevets protègent les idées qui sont derrière, et servent effectivement à éliminer toute compétition. En brevetant par exemple le format des documents produits par un logiciel, un éditeur pourrait interdire à tout autre programme de lire ce format, et empêcher ainsi les utilisateurs d'accéder à leurs propres données sans passer par son programme. Stéphane Fermigier, président de l'AFUL, note que «*les brevets sont donc un frein très fort à l'interopérabilité, et ne peuvent qu'aboutir à un renforcement ou à l'extension de positions monopolistiques déjà existantes*». Certains acteurs industriels font actuellement pression pour que l'Europe adopte le même système, mais les développeurs et les responsables de PME européennes dans le domaine de l'informatique et des télécommunications sont pour la plupart opposés aux brevets logiciels, comme en témoignent les 80'000 signatures recueillies par la pétition pour une *Europe sans brevets logiciels* [23].

Tout cela montre que convaincre et essayer d'imposer GNU/Linux et les logiciels libres dans l'entreprise ou les administrations n'est pas facile. L'histoire de l'informatique et de la technique en général a montré plu-

sieurs fois que ce n'est pas forcément le meilleur produit qui sera plébiscité, mais que ce sera souvent celui qui aura le poids financier, juridique et marketing important, ou qui arrivera le premier sur le marché. Mais alors quels sont les arguments qui pourraient faire changer le cours de l'histoire?

LE COURS DE L'HISTOIRE

Les arguments décisifs pourraient être la venue dans le monde du logiciel libre de gros calibres tels que IBM, Oracle, Sun ou HP, dont la seule présence donne une impression de crédibilité, et par les volontés politiques comme c'est le cas dans plusieurs pays comme la France, l'Allemagne, le Mexique ou la Chine.

Les récentes déclarations de Microsoft, en particulier celles de Craig Mundie, Senior Vice President, dans **The Commercial Software Model** [26], ainsi que des documents internes qui avaient été révélés au public en octobre 1998 (les célèbres documents *Halloween* [21]) tendent à montrer une certaine nervosité de la part du géant égratigné et laissent à penser que quelque chose est bien en train de changer. Mais si l'histoire doit nous servir de guide, signale Stéphane Bortzmeyer dans son article *After Word: l'avenir du traitement de texte* [40], alors c'est le changement de paradigme qui bouleversera cette situation monopoliste. *Le BUNCH (l'ensemble des concurrents officiels d'IBM dans les années 70) n'a pas réussi à ébranler la domination de Big Blue. Seuls ceux qui n'ont pas fait de l'IBM, c'est-à-dire Digital, Apple, Commodore ou Atari, ont ramené un marché monopoliste à un marché concurrentiel, en ouvrant les portes de la micro-informatique.*

GNU/Linux et les Logiciels Libres sont un nouveau paradigme. Ils ouvrent les portes à un nouveau modèle économique qui utilise le réseau Internet comme base de travail et le *bazar* organisé pour produire des outils d'une grande qualité. Les chiffres sont là pour montrer que ce nouveau modèle économique fonctionne puisque les logiciels libres phares sont très utilisés. Par exemple le serveur de nom de domaine (DNS) *bind* repré-

sente 90% des serveurs aux USA, les trois serveurs de mail *sendmail*, *postfix* et *gmail* représentent 70% et le plus connu, le serveur Web Apache plus de 60% des serveurs installés dans le monde. Une étude de l'IDC montre que la part de marché de Linux croît plus vite que prévu. Il était projeté qu'il serait numéro 2 en 2002 ou 2003 et c'est arrivé en 1999. Une autre étude de Forrester Research montre que 56% des grands groupes mondiaux utilisent des logiciels libres. Un autre signe tangible est le fait que Linux remplace petit à petit les Unix propriétaires tels que IRIX de Silicon Graphics, ou AIX d'IBM. Hewlett-Packard a de son côté engagé Bruce Perens, un des défenseurs de longue date des systèmes libres et ouverts, en tant que *conseiller pour les stratégies Open-Source* et fera peut-être bientôt le pas. Si la venue du monde *économique* dans les logiciels libres est souhaitable pour donner une image plus *professionnelle*, on peut s'interroger sur les raisons qui poussent ces entreprises à investir des sommes parfois considérables (1 milliard de dollars pour IBM en 2001) pour le développement d'applications *libres* ou *Open-Source*.

UN MODÈLE ÉCONOMIQUE – UN NOUVEAU PARADIGME

Les motivations d'acteurs comme Sun ou SAP pour se lancer dans le monde du libre sont multiples, mais la principale est certainement la volonté de contrer la concurrence dans un marché qui tend au monopole. Ainsi, les entreprises minoritaires dont les produits ont une part de marché négligeable ont intérêt à les développer de façon ouverte, dans l'espoir d'en augmenter la diffusion et d'en tirer partie ensuite pour développer d'autres services à valeur ajoutée. Sur ce principe, SAP a ouvert le code de sa base de données SAPdb, espérant créer une référence dans le domaine à partir d'un logiciel qui était tombé dans l'oubli.

Mais ce principe est aussi valable dans le cas d'un développement en interne que l'on souhaite maintenir pour ne pas le laisser tomber dans l'oubli. C'est la voie qu'a choisie *Matra division*, en ouvrant le code d'*OpenCascade* [11], son modèleur

3D, qui représente plusieurs dizaines d'années-hommes de travail et un investissement de 75 millions d'euros. Dans un schéma conventionnel, cet outil serait probablement tombé dans l'oubli. Suite à l'ouverture de ce logiciel, une spin-off a été créée, générant des revenus grâce aux services fournis autour du logiciel, et comptant déjà 130 clients dans 17 pays.

Il ne faut pas oublier que les informaticiens payés pour produire des logiciels qui seront disponibles sur le marché représentent tout juste 15%. 85% des lignes de code *écrites* tous les jours ne le sont qu'à des fins internes, et leur ouverture ne menacerait pas les activités lucratives de ces entreprises. Bien entendu, dans certaines applications particulières, c'est le savoir-faire de ces entreprises qui est contenu dans ces programmes, et le diffuser sans contrepartie serait suicidaire, mais la plupart du temps, les entreprises ont les mêmes besoins, et réinventent la roue, alors même que l'on manque d'informaticiens. Les logiciels libres permettent dans ces cas de mutualiser ces développements, et c'est ce que propose par exemple Sourceforge (23'000 projets, 200'000 développeurs enregistrés) qui permet un travail collaboratif des développeurs à travers Internet.

Les Logiciels Libres et la large diffusion sont la meilleure méthode pour imposer un standard s'il est accepté par la communauté. De tels standards sont publiés, et favorisent la compatibilité entre les logiciels, au contraire des standards propriétaires, qui sont généralement conçus pour retenir l'utilisateur en l'empêchant d'utiliser d'autres logiciels que ceux du concepteur du *standard*. L'exemple le plus flagrant est Internet, qui est entièrement basé sur des standards libres et qui n'aurait jamais pu s'étendre comme il l'a fait s'il avait été conçu sur la base de standards propriétaires ne fonctionnant qu'avec certains systèmes. A l'inverse, la plate-forme .NET projetée par Microsoft tente de construire un service réseau supplémentaire, fermé, au dessus des protocoles d'Internet, afin d'empêcher les autres logiciels d'y accéder.

Comme on peut le voir il existe des arguments intéressants et un potentiel sur le plan commercial qui plaident en faveur des logiciels libres. On

déplace ainsi les revenus sur les services, et sur ce principe le client a beaucoup à y gagner, puisque le service est justement l'élément principal du *business model* de l'Open Source. Alain Lefebvre, vice-président du groupe SQLI, souligne que «*nous entrons dans l'ère du service après avoir vécu successivement l'ère du matériel puis l'ère du logiciel*».

Mais si on veut bouger l'ordre établi, il faut secouer l'immense pyramide de l'immobilité et donner l'exemple. C'est à ce stade que les gouvernements et les personnalités politiques se doivent d'intervenir.

UN ENGAGEMENT POLITIQUE

Dans un certain nombre de pays, l'engagement politique sur le sujet devient de plus en plus présent. Ainsi, le rapport du député français Thierry Carcenac **Pour une Administration Electronique citoyenne [16]**, adressé au premier ministre, souligne que «*pour un certain nombre de tâches, les logiciels open source se sont révélés fiables, performants, sécurisés, compétitifs financièrement*». Les propositions de ce rapport sont de mettre l'ensemble des développements réalisés par ou pour les administrations sous licence *Open Source* ou analogue, ainsi que d'avoir une distribution de base pour l'administration. Et de rajouter que dans une utilisation gouvernementale, le caractère ouvert et public du code source des logiciels libres permet d'en améliorer la pérennité et la sécurité. Par ailleurs, le caractère coopératif du développement permet des tests très nombreux et assure ainsi la robustesse des logiciels produits. Le forum qui a suivi ce rapport a donné lieu à un large débat dans lequel les contributeurs ont manifesté, dans l'ensemble, une forte appétence pour le renforcement de la place des logiciels libres dans les systèmes d'information des administrations.

Lors de la conférence sur les logiciels libres à l'EPFL, le 12 juin dernier, Jean-Pierre Archambault, actuellement à la *Mission Veille technologique* du Centre National de Documentation Pédagogique en France (CNDP), a fait part des récentes prises de positions de l'Etat français en

faveur des logiciels libres et a montré que la tendance était amorcée et qu'il existait une réelle volonté de la part des responsables d'établissements et d'administrations publiques. Pour ne citer que quelques exemples, le ministère de la culture a commencé la migration de 400 serveurs sous GNU/Linux, et la Direction Générale des Impôts, très attentive aux problèmes liés à la sécurité, a choisi ce même système pour ses 950 serveurs. La Chine ou le Mexique ont quant à eux adopté des mesures radicales, en passant respectivement tout l'appareil administratif sous Linux *Red Flag* ou toutes les écoles, soit, 150'000 établissements, sous GNU/Linux.

En Suisse aussi, le logiciel libre fait son apparition dans les administrations. Ainsi, à Genève, quelques enseignants ont lancé des expériences d'utilisation de logiciels libres, que ce soit pour la partie *enseignement* (avec StarOffice), ou pour l'administration (serveurs de fichiers utilisant Samba et bases de données libres pour la gestion des notes). Tout récemment, un serveur GNU/Linux utilisant Apache et Perl a été mis en place au Palais de Justice genevois pour permettre l'accès aux fiches de jurisprudence genevoise [15].

Dans les autres pays on découvre que la volonté politique de promouvoir les logiciels libres est souvent liée aux problèmes de sécurité nationale. Ainsi, pour l'Allemagne, la France ou les Etats-Unis, il est tout simplement inconcevable de ne pas maîtriser de bout en bout leurs systèmes d'informations sensibles (militaires en particulier), et ceci est impossible à réaliser avec des programmes propriétaires. Les allemands ont été les premiers à réagir en supprimant tous *les produits Microsoft des postes sensibles* [28] dans le ministère des Affaires étrangères et celui de la Défense. Le dernier rapport de la Communauté Européenne [24] sur le réseau d'écoute Echelon *invite les institutions européennes et les administrations des états membres à promouvoir des projets de logiciels dont le texte-source soit publié, étant donné qu'il s'agit là de la seule manière de garantir qu'ils ne comportent pas de backdoors (défauts cachés)*.

Peut-être plus important encore, la démocratie est aussi impliquée. Des controverses ont surgi en Belgique sur

l'utilisation de systèmes propriétaires pour gérer les **votes électroniques**, un domaine où la transparence devrait être la règle absolue dans une démocratie. Des procédures sont en cours devant des tribunaux belges, des citoyens ayant argumenté qu'il leur était impossible d'être convaincus de la transparence d'un scrutin et de l'absence de trucage, si en parallèle on refusait de leur laisser vérifier le système de vote. L'utilisation de logiciels ouverts, permettant à n'importe qui (s'il en a les compétences) de vérifier le fonctionnement du système est donc indispensable pour éviter que le processus de vote ne soit entre les mains de quelques techniciens seulement. Le gouvernement genevois est malheureusement en train de suivre la mauvaise voie avec un projet pilote de vote par Internet, dont la réalisation a été confiée à deux entreprises commerciales et pour laquelle des indices semblent montrer qu'une solution entièrement propriétaire, sur lesquels les citoyens n'auront aucun contrôle direct, sera choisie.

L'AVENIR ...

On découvre donc que le monde du logiciel bouge à différents niveaux. Avec l'adoption des Nouvelles Technologies de l'Information et de la Communication (NTIC), étape décisive pour de nombreuses entreprises, il faut que l'on puisse garantir l'interopérabilité, la pérennité et la sécurité des systèmes d'information. Après le passage douloureux de l'an 2000, les diverses migrations et les mises à jour très coûteuses des systèmes, les ingénieurs et les décideurs hésitent de plus en plus à suivre l'évolution effrénée du matériel informatique et des logiciels. Les changements sont trop rapides et les stratégies changent vite, aujourd'hui Novell demain Lotus Notes, après demain Windows XP et pourquoi pas le leasing des logiciels et la technologie décentralisée .NET le mois prochain? Une réflexion de fond doit s'établir et ce qui est sûr aujourd'hui, c'est que l'on a des nouveaux paramètres et arguments à prendre en compte. Comme le souligne Bill Gates, *«cet écosystème dans lequel cohabitent des logiciels libres et des logiciels commerciaux est essentiel et très sain*

car les utilisateurs ont toujours le choix». Peut-être faudrait-il plutôt dire à *nouveau le choix*.

Pour être objectif, les logiciels libres sont à l'heure actuelle une solution pour les entreprises dans le domaine des serveurs, où les systèmes d'exploitation libre GNU/Linux, FreeBSD ou OpenBSD sont souvent cités en exemple. Il n'en est pas de même en ce qui concerne les postes de travail où les applications métiers telles que la CAO, DAO, la comptabilité ou la gestion de stock n'ont pas encore été portées sur ces plates-formes libres. Même si des suites bureautiques comme StarOffice – <http://www.staroffice.org> – de Sun Microsystems ou KOffice de KDE – <http://www.kde.org> – sont une alternative de qualité à la suite Office de Microsoft, les secrétaires ne sont pas encore prêtes à se passer de cette dernière. Le jour où l'on privilégiera le format des données aux applications qui les manipulent, alors seulement on aura fait un grand pas vers la compatibilité et la pérennité des données. Le mouvement du logiciel libre a prouvé qu'il était très dynamique, une source de nouvelles idées, d'innovations et de partage. Même si les logiciels libres n'ont de loin pas fini de se développer, ils ne remplaceront certainement pas entièrement les produits propriétaires, qui ont de fortes compétences, en particulier dans l'approche métier, et il est probable que les deux mondes coexisteront, chacun se concentrant sur les domaines où il est le meilleur.

La plupart des grandes découvertes de ce monde et leurs implications sont dues aux travaux collaboratifs et à la libre circulation de la connaissance. Même dans la recherche médicale, où la concurrence fait rage, les collaborations entre les instituts de recherche sont nécessaires. Comme le dit Roberto di Cosmo, professeur d'informatique à l'université de Paris VII, *«les logiciels libres ne sont pas seulement une bonne idée, ils sont aussi une nécessité»*. Si les logiciels libres sont une nécessité pour l'entreprise et pour l'économie en général, ils ont aussi un rôle important à jouer dans l'éducation, où les notions de pluralité, liberté et égalité sont fondamentales, mais ceci nécessiterait l'ouverture d'un autre débat et d'un article à lui tout seul.

RÉFÉRENCES

GROUPES ET SITES GÉNÉRAUX

- [1] **GULL**, Groupe des Utilisateurs Linux et des Logiciels Libres du Léman, <http://www.linux-gull.ch>
- [2] **AFUL**, Association Francophone des Utilisateurs de Linux et des Logiciels Libres, <http://www.aful.org>
- [3] **FSF**, Free Software Foundation, <http://www.fsf.org>
- [4] **APRIL**, Association Pour la Promotion et la Recherche en Informatique Libre, <http://www.april.org>

ENTREPRISES DE SERVICES SPÉCIALISÉES DANS LES LOGICIELS LIBRES

- [5] **Goelaan**, société romande, <http://www.goelaan.ch>
- [6] **ProLibre**, société romande, <http://www.prolibre.com>
- [7] **Alcôve**, société française, <http://www.alcove.com>
- [8] **Easter-Eggs**, société française, <http://www.easter-eggs.com>
- [9] Recherches dans la liste des entreprises inscrites au GULL, <http://www.linux-gull.ch/cgi-bin/admin.pl>

LOGICIELS ET PROJETS LIBRES, ET EXEMPLES D'UTILISATION

- [10] **Apache**, serveur Web libre utilisé sur 60% des serveurs Web dans le monde, <http://www.apache.org>
- [11] **OpenCascade**, modèleur 3D de Matra Division, <http://www.opencascade.org>
- [12] **Samba**, logiciel permettant à un serveur de partager des fichiers et des imprimantes avec un client Windows, <http://www.samba.org>
- [13] **Sourceforge**, services pour le développement d'applications Open Source, <http://www.sourceforge.org>
- [14] Ferme de machines sous GNU/Linux pour le calcul de simulations astronomiques à l'Observatoire de Genève, <http://obswww.unige.ch/~pfennige/gravitor/gravitor.html>
- [15] Fiches de jurisprudence genevoise, un serveur utilisant GNU/Linux, Apache et Perl au Palais de Justice genevois, <http://justice.geneve.ch/jurisprudence/>

ÉTUDES ET RAPPORTS

- [16] Le Rapport **Carcenac**, *Pour une administration électronique citoyenne - Méthodes et Moyens*, Rapport remis au Premier ministre par Thierry Carcenac, député du Tarn, le 19 avril 2001, <http://www.internet.gouv.fr/francais/textesref/rapcarcenac/sommaire.htm>
- [17] Résumé de l'étude de SWePIX, Swiss Web Performance IndeX, <http://www.sysformance.com/d/news/2000.html#Anchor-54790>

- [18] Netcraft Web Server Survey, statistiques d'utilisation des différents serveurs Web du marché, <http://www.netcraft.co.uk/survey/>
- [19] Journée sur les logiciels libres du 6 octobre 2000, organisée par l'Observatoire Technologique de l'Etat de Genève et le GULL, http://www.geneve.ch/obstech/Manifestations/rencontres.html#Logiciels_libres
- [20] Linux in Retail & Hospitality: *What Every Retailer Should Know*. Microsoft Corporation, White Paper, février 2001, http://www.microsoft.com/europe/retail/Multi_channel_retail/364.htm
- [21] Documents Halloween, Rapports internes de Microsoft sur les logiciels libres, <http://www.opensource.org/halloween/>
- [22] Page d'information sur le e-voting à Genève, sur le site du GULL, <http://www.linux-gull.ch/evote/>
- [23] La Pétition pour une Europe sans brevets logiciels d'EuroLinux, <http://petition.eurolinux.org/>
- [24] Rapport de la Communauté Européenne sur le réseau d'écoute Echelon, préconisant l'emploi de logiciels ouverts, http://www.europarl.eu.int/tempcom/echelon/pdf/prechelon_fr.pdf
- [25] Security focus, site répertoriant les problèmes de sécurité informatique, <http://www.securityfocus.com/>
- [26] **Bernard Lang**, directeur de recherche à l'INRIA, *Logiciels Libres et Entreprises*, 25 sept. 99, <http://pauillac.inria.fr/~lang/ecrits/monaco/>
- [27] **Eric S. Raymond**, *The Cathedral and the Bazaar*, http://www.firstmonday.org/issues/issue3_3/rammond/index.html, et Traduction française de Sébastien Blondeel, http://www.linux-france.org/article/these/cathedrale-bazar/cathedrale-bazar_monoblock.html, une description des méthodes de développement des logiciels ouverts, et comparaison avec les méthodes commerciales.
- [28] **Bruce Schneier**, *Open Source and Security*, Cryptogram, sept. 99, pourquoi les codes sources doivent être ouverts afin d'améliorer la sécurité, <http://www.counterpane.com/crypto-gram-9909.html#OpenSourceandSecurity>
- [29] **Bruce Schneier**, *An Intentional Back Door*, Cryptogram, fév. 01, sur la *backdoor* du logiciel Interbase de Borland, <http://www.counterpane.com/crypto-gram-0102.html#5>
- [30] **David A. Wheeler**, *Is Open Source Good for Security?*, *Secure Programming for Linux and Unix HOWTO*, <http://www.linuxdoc.org/HOWTO/Secure-Programs-HOWTO/open-source-security.html>
- [31] **Natalie Walker Whitlock**, *Does open source mean an open door?*, Casaflora Communications, mars 01, <http://www-106.ibm.com/developerworks/linux/library/l-oss.html>
- [32] *Open source software: Will it make me secure?*, par John Viega, Reliable Software Technologies, <http://www-106.ibm.com/developerworks/security/library/oss-security.html>
- [33] *Google keeps pace with demands*. Détails techniques de l'infrastructure de Google, <http://www.internetteek.com/infrastructure01/infra050701.htm>
- [34] *After Word: l'avenir du traitement de texte*, par Stéphane Bortzmeyer, <http://www.internatif.org/bortzmeyer/afterword/afterword.html>
- [35] *L'Open Source pousse les éditeurs à l'abandonware*, par Alain Lefebvre, janv. 01, http://solutions.journaldunet.com/0101/010111_decrypt_oss.shtml
- [36] *Pourquoi les projets Open Source sont supérieurs*, par Alain Lefebvre, fév. 01, <http://solutions.journaldunet.com/0102/010207decrypt.shtml> ■

ARTICLES ET PUBLICATIONS

- [26] **Craig Mundie**, Senior Vice President de Microsoft, *The Commercial Software Model*, <http://www.microsoft.com/presspass/exec/craig/05-03sharedsource.asp>
- [27] *Microsoft, Linux et la vague open source...*, interview de **Bill Gates** dans CNet, traduite par ZDNET.fr, <http://news.zdnet.fr/story/0,,s2089671,00.html>
- [28] *Bundeswehr verbant Microsoft-Programme*, Der Spiegel, 17 mars 01, <http://www.spiegel.de/netzwelt/politik/0,1518,123170,00.html>
- [29] **Roberto di Cosmo**, *Le prix de l'amour (électronique)*, 9 mai 00, sur les virus s'attaquant aux applications Windows, <http://www.pps.jussieu.fr/~dicosmo/LovePrice.html>
- [30] **Roberto di Cosmo**, *Piège dans le Cyberspace*, sur le monopole de Microsoft et les procédés de fidélisation des clients, <http://www.pps.jussieu.fr/~dicosmo/Piege/PiegeFR.html>
- [31] Sécurité, informatique et vie privée, présentation à la journée sur les logiciels libres, sur les problèmes de sécurité et les dangers pour la vie privée que posent les logiciels propriétaires, <http://www.pps.jussieu.fr/~dicosmo/TALKS/OT-Geneve-10-2000/>

Libre reproduction © Copyright Gilbert Robert et Frédéric Schütz- GNU Free Documentation License .

INTÉRÊT GÉNÉRAL ET PROPRIÉTÉ INTELLECTUELLE

Philippe QUÉAU, DIRECTEUR DE LA DIVISION DE LA SOCIÉTÉ DE L'INFORMATION À L'UNESCO



Le droit ne descend pas tout armé de la cuisse de Jupiter. Il est en général conçu par ceux qui disposent du pouvoir et qui légifèrent au nom de l'intérêt général. Mais le même droit peut produire des effets très différents suivant la place qu'on occupe dans la société. Comme le note Anatole France: «*Notre droit défend au riche comme au pauvre, dans une majestueuse égalité, de voler du pain et de mendier au coin des rues.*» Le droit est une «*machine de guerre en faveur des plus riches*» surenchérit François Dagognet¹. Il n'est pas interdit de penser que cette faveur envers les riches et les forts est aujourd'hui en passe de s'aggraver.

Ce biais s'observe en particulier dans le domaine, central à l'âge du virtuel, de la propriété intellectuelle.

Le droit actuel de la propriété intellectuelle emprunte beaucoup de sa philosophie au droit de la propriété matérielle. Le droit a toujours eu tendance à appliquer aux productions de l'intellect des conceptions proches de celles régissant la propriété des biens réels. Cette tendance est ancienne et correspond à une sorte de mouvement de fond. Ainsi le droit d'auteur est basé sur l'idée que les productions de l'esprit sont éminemment personnelles, liées à des personnes bien réelles, qui sont les concepteurs, les auteurs, les inventeurs.

Mais dans le même temps, les législateurs ont également mis l'accent sur les besoins de la société de bénéficier collectivement de cette richesse intellectuelle, en vue de l'intérêt général. C'est pourquoi des dispositions variées sont apparues pour limiter et réduire les droits de propriété intellectuelle, avec en vue la préoccupation du bien commun.

Il s'agissait de trouver un équilibre entre deux notions essentiellement opposées: le besoin de la société d'accéder au savoir, à la création, aux résultats de l'invention humaine, et les droits de l'individu créateur.

Revenons aux fondements. Il s'agit avant tout de protéger l'intérêt général et le progrès des sciences et des arts, en assurant la diffusion universelle des connaissances et

*Il en est des livres comme du feu de nos foyers,
on va prendre le feu chez son voisin,
on l'allume chez soi, on le communique à d'autres
et il appartient à tous.*

Voltaire

des inventions, en échange d'une protection consentie par la collectivité aux auteurs pour une période limitée.

C'est une intuition première. Il est plus avantageux pour l'humanité de faire circuler librement les idées et les connaissances que de limiter cette circulation. Aristote affirme que l'homme est l'*animal mimétique* par excellence. Pour Condillac, «*les hommes ne finissent par être si différents, que parce qu'ils ont commencé par être copistes et qu'ils continuent de l'être.*» Pour le philosophe Alain, «*copier est une action qui fait penser.*» Thomas Jefferson, rédacteur de la constitution des Etats-Unis, promoteur du concept de bibliothèque publique et de la doctrine du *fair use* permettant les citations de textes protégés à des fins éducatives ou de recherche, écrivait aussi: «*les inventions ne peuvent pas, par nature, être sujettes à la propriété.*»

Cette intuition philosophique fondamentale a d'ailleurs guidé le législateur. Si la société consent à reconnaître un droit de propriété intellectuelle à l'inventeur d'un procédé, c'est pour en éviter la perte ou l'oubli, pour en faciliter la description publique, et pour en autoriser après une période de protection limitée la libre copie (encourageant de ce fait la concurrence).

Si l'on cherche à protéger l'auteur, c'est pour lui assurer un revenu, et pour protéger son *droit moral*. Mais la création d'un monopole sur l'exploitation des œuvres après la mort de l'auteur, et l'extension de ce monopole, n'est pas en soi de nature à favoriser la création. Elle aurait plutôt tendance à inciter les éditeurs à vivre sur leur catalogue d'auteurs reconnus, plutôt que d'encourager la recherche de nouveaux talents.

Or, depuis quelque temps, on assiste à des modifications du droit de la propriété intellectuelle qui semblent favoriser certains intérêts sectoriels et qui remettent donc en cause l'équilibre ancien entre ayants droit et utilisateurs.

Par exemple, en Europe, la directive du 29 octobre 1993 relative à l'harmonisation de la durée de protec-

¹ François Dagognet, Philosophie de la propriété, Paris 1992

tion du droit d'auteur a allongé la protection des œuvres littéraires de 50 à 70 ans après la mort de l'auteur.

Depuis le début du siècle, le Congrès américain allonge régulièrement la durée du *copyright* au détriment du domaine public. En 1998, le 27 octobre, le Congrès a voté le **Sonny Bono Copyright Term Extension Act** qui fait passer la durée du *copyright* de 50 à 70 ans après la mort de l'auteur, et de 75 à 95 ans pour les droits des entreprises. On peut interpréter cet allongement unilatéral de la protection, qui revient à empiéter sur le *domaine public*, comme faisant essentiellement le jeu des grands groupes de communication, et on pourrait y voir aussi une tendance à la disparition pure et simple du domaine public.

Cette évolution voulue par les éditeurs – et obtenue sans réel débat démocratique — est incompatible avec le développement d'un accès universel à l'information et contraire à l'esprit même de la loi sur la propriété intellectuelle. La collectivité accepte en effet de reconnaître et de protéger les droits exclusifs des créateurs sur leurs œuvres, mais pour une durée limitée seulement, étant entendu que ces œuvres doivent *in fine* revenir à la collectivité, et satisfaire ainsi à l'intérêt général, qui est d'encourager la libre circulation des idées et l'accès de tous aux connaissances.

Cette tendance de fond à renforcer (sans contrepartie pour l'intérêt général) les intérêts catégoriels peut aussi se lire à travers l'évolution du droit de la propriété intellectuelle sur le vivant ou sur les OGM (organismes génétiquement modifiés).

L'évolution récente du droit de la propriété intellectuelle va-t-elle réellement dans le sens de l'intérêt général? Par exemple, le problème des bases de données et des données du secteur public fait l'objet d'un débat qui reste toujours ouvert², opposant en gros certains pays développés au reste du monde.

Une bataille historique à cet égard s'est tenue en décembre 1996 à Genève, lors de la **conférence diplomatique sur certaines questions de droits d'auteur et de droits voisins**, organisée par l'Organisation Mondiale de la Propriété Intellectuelle. Pour résumer l'ampleur des critiques et des craintes qu'ont suscité ces propositions, citons la Fédération internationale d'Information et de Documentation: «*le rôle des collecteurs et des diffuseurs publics d'information (bibliothèques, archives, musées,...) pourrait être détruit*». Ou encore la réaction de l'IFLA (Fédération Internationale des Associations de Bibliothèques): «*Ces propositions vont obstruer plutôt qu'améliorer le flot des informations... La tendance actuelle à la protection des droits d'auteur pour des raisons purement écono-*

miques semble être en conflit avec le but original du copyright de promouvoir le progrès des sciences et des arts.»

Le traité sur les bases de données proposait la création d'un nouveau droit de propriété intellectuelle, le droit dit *Sui Generis*, suscitant des réactions extrêmement violentes de la part de représentants de la communauté scientifique internationale, comme le Conseil international pour la Science (ICSU)³.

Ainsi le public pourrait être amené à être obligé de payer pour disposer d'informations du domaine public.

La question de la propriété des données brutes est particulièrement préoccupante à un moment où l'Etat se *désengage*, et sous-traite beaucoup de ses bases de données à l'industrie privée pour les gérer. Les informations contenues dans ces bases appartiennent de plein droit au domaine public. L'Etat ayant le monopole de la collecte de ces informations publiques, il ne saurait s'en désintéresser sans préjudice pour le citoyen. Les sous-traitants privés qui gèrent ces bases de données publiques ne devraient pas devenir de ce fait *propriétaires* des données elles-mêmes, ou ce qui revient au même, du droit exclusif d'en disposer.

Autre idée fondamentale: on ne peut pas protéger une idée pure. Par exemple un principe, une méthode ou un système de pensée, un théorème, une théorie ou un algorithme. On peut en revanche protéger un logiciel informatique particulier qui utiliserait tel principe ou tel algorithme.

Evidemment cette doctrine ouvre la voie à d'innombrables contestations. Car comment tracer précisément la frontière entre le réel et le virtuel, surtout quand il s'agit d'inventions de plus en plus dématérialisées? On se rappelle la bataille du *look and feel* opposant Apple à Microsoft sur l'*apparence* de l'interface de leurs systèmes d'exploitation.

Si on prenait un peu de recul, on pourrait inférer l'hypothèse qu'avec chaque innovation technologique, le droit a toujours eu tendance à renforcer (en théorie) le droit de la propriété intellectuelle, et ce d'autant plus que l'évolution technologique rendait d'autant plus obsolète (en pratique) la philosophie même de ce droit.

Aujourd'hui l'enjeu du droit de la propriété intellectuelle est devenu stratégique. Il s'agit de la source principale de production de richesses dans le cadre de la *société de l'information*. Les *lobbies* concernés s'agitent. Ils demandent d'étendre et de renforcer les droits sur les marques, sur les brevets, sur les bases de données, et plus généralement sur tout ce qui pourrait faire figure d'activité intellectuelle. Les intérêts du public au sens large ne sont pas représentés. Pas d'avocats, pas de *lobbyistes* dans les couloirs de Bruxelles ou de Washington, pour défen-

2 Cette question est débattue dans le livre vert de la Commission européenne: «*L'information émanant du secteur public: une ressource clef pour l'Europe – Livre vert sur l'information émanant du secteur public dans la société de l'information*» COM (1998) 585.

3 «*The EU Directive could irreparably disrupt the full and open flow of scientific data which ICSU has long labored to achieve, and could seriously compromise the worldwide scientific and educational missions of its member bodies. (...) All data – including scientific data – should not be subject to exclusive property rights on public policy grounds.*»

dre les intérêts des utilisateurs. L'intérêt général, une vision supérieure du *bien commun* n'apparaissent pas clairement, et les militants de cette cause ne sont pas légion.

Les besoins des écoles ou des universités, les difficultés des pays les moins avancés ou des milieux défavorisés des pays développés, ne reçoivent pas d'attention particulière dans le contexte de cette intense activité de *lobbying*.

De fait, on observe une extension spectaculaire des droits de propriété intellectuelle, et parfois dans des directions inattendues. Par exemple, on a réussi à faire admettre le principe d'une privatisation du génome humain aux États-Unis. Ceci se passe dans une relative indifférence de la part du public, très mal informé des conséquences lointaines de tels hold-up ou *land grab* sur le patrimoine de l'humanité.

Pourtant des voix dissonantes tentent de se faire entendre. Par exemple, des développeurs de logiciels s'inquiètent. Ils ne veulent pas que leurs propres productions logicielles soient trop protégées. Ils estiment que cela irait contre leur intérêt. Cela aurait pour conséquence de créer des oligopoles surpuissants, éliminant toute concurrence possible, avec à la clé une diminution de la création et de l'invention.

Le durcissement du cadre juridique protégeant la propriété intellectuelle peut sembler, *a priori*, en faveur des créateurs et des innovateurs actuels. Mais en réalité, sur le long terme, il est très vraisemblable que les créateurs de demain et les futurs innovateurs se retrouvent prisonniers d'un système de *clôture intellectuelle* empêchant toute citation, réutilisation, inspiration, évocation, parodie, si on laisse ce gigantesque *land grab* s'opérer sans contre-pouvoir. Car rien ne sort jamais de rien. Aucun créateur ne crée *ex nihilo*. Le critique littéraire Northrop Frye disait qu'«*on ne peut faire de la poésie qu'avec d'autres poèmes, et des romans qu'avec d'autres romans*». On pourrait dire la même chose des programmes informatiques ou de la mise au point de nouvelles molécules. Si on limite l'accès aux informations et aux connaissances produites par la société, par exemple en empêchant le recours normal aux exceptions légales permettant le droit de citation et le droit de copie à des fins d'enseignement et de recherche, alors ce ne sera pas *l'encouragement des sciences et des arts* qui en résultera, mais bien au contraire la création de monopoles indéradicables, le renforcement de la fracture numérique.

Ceci est vrai de toutes les formes d'activité intellectuelle. Où se situe l'intérêt général? Il s'agit d'encourager

la création et l'invention, et donc d'encourager l'éducation, la recherche et l'accès le plus large possible aux ressources informationnelles. Si les professeurs ne peuvent plus citer les textes, si les étudiants ne peuvent plus les copier, si on laisse les bureaux d'enregistrement des brevets octroyer des brevets pour l'*invention* de procédés parfaitement anodins (ainsi le brevet accordé à Amazon pour l'achat en ligne en une seule opération, dite *one-click*, qui lui a permis de poursuivre son compétiteur Barnes and Noble en justice), ou pour des idées triviales (comme les brevets accordés à Dell Computer autour de l'*idée* de vendre des ordinateurs en ligne), alors les futurs créateurs et inventeurs seront empêchés de développer de meilleurs modèles, de tester des améliorations successives. Il est absolument nécessaire que le système *mondialisé* de la propriété intellectuelle prenne réellement en compte – également au plan mondial – les intérêts contradictoires en présence. Les pays du Nord n'ont pas à cet égard forcément les mêmes intérêts que les pays du Sud. Mais dans les pays développés aussi, la ligne de contradiction interne est

très complexe à tracer, entre utilisateurs et ayants droit, entre actionnaires, consommateurs et citoyens.

Ce qui est sûr, c'est qu'un système équilibré de propriété intellectuelle doit garantir l'existence d'un vaste *domaine public*, librement accessible à tous à des fins d'usage personnel, d'enseignement ou de recherche.

Ce domaine public doit être protégé contre toute forme de privatisation. Il existe d'excellentes

techniques juridiques à cet effet comme le *copyleft* et la *Licence Publique Générale (General Public License – GPL)* développée par Richard Stallman et appliquée notamment à la protection de LINUX contre toute tentative d'appropriation partielle ou non par des entreprises privées.

Il faut poser le problème de la propriété intellectuelle selon des prémisses complètement différentes, tenant compte de l'émergence progressive de la civilisation du virtuel. Il est sans doute temps d'inventer un nouveau modèle de législation, et mieux: une nouvelle pratique. Il faut trouver de nouvelles manières de rétribuer l'activité inventive, il faut s'efforcer de déterminer de manière plus fine les responsabilités quant à la protection des véritables créateurs, et éviter que des avantages indus soient accaparés par des opérateurs déjà surpuissants, et qui pourraient, si l'on n'y prend garde, acquérir des pouvoirs inacceptables.

La discrimination entre les idées (non protégeables)



et l'expression de ces idées (théoriquement protégeable) devient de plus en plus difficile. L'expression des idées peut prendre de nombreuses formes matérielles essentiellement équivalentes, qu'il est aisé de détourner, de modifier matériellement à des degrés très divers.

Devant ces difficultés de plus en plus pressantes, la jurisprudence a eu tendance à affaiblir la portée des lois sur la propriété intellectuelle dans les domaines les plus exposés à ces contradictions. Le *Look and feel*, que nous avons déjà évoqué, n'est pas actuellement protégeable, par exemple. Mais les difficultés sont innombrables. Un brevet fut accordé en août 1993 à Compton's NewMedia, une division de Tribune Company, sur des fonctions d'indexation pouvant s'appliquer aussi bien aux CD-ROM qu'aux services en ligne. Le brevet décrivait un *système de recherche multimédia utilisant plusieurs index capables de montrer le degré de corrélation des informations*. Il fut envisagé par l'entreprise de collecter des royalties au salon technique Comdex de novembre 1993 sur la base de ce brevet très généreux. La légitimité du brevet fut évidemment contestée par toute l'industrie concernée, et le brevet fut *réévalué* par le US Patent and Trademark Office puis annulé en 1994.

Si la communauté des industriels du secteur a pu réagir avec promptitude au caractère outrancier du brevet ainsi trop légèrement accordé, il est à craindre que des communautés moins mobilisables et moins informées, comme le grand public, soient soumises à des décisions juridiques très désavantageuses, privilégiant des groupes de pression mieux outillés pour faire prévaloir leurs points de vue.

La révolution en cours va si loin que l'équilibre classique entre auteurs, intermédiaires (éditeurs, diffuseurs) et utilisateurs va certainement être affecté dans un sens ou dans un autre. C'est le fonctionnement social des échanges d'informations et d'idées qui pourrait être remis en cause, ce qui pourrait avoir pour conséquence la disparition de droits acquis depuis longue date comme le concept de bibliothèque publique ou l'usage des œuvres pour l'éducation et la recherche (*fair use*).

Les connaissances acquises par l'humanité forment en principe un bien public mondial, appartenant à tous. «*Le naturel, ainsi que le rationnel, ce que la nature nous donne ou implique ne sauraient être réservés à quelques-uns.*»⁴ En pratique, certains savent en tirer un avantage comparatif particulièrement décisif, et beaucoup d'autres, qui gagneraient immensément à avoir accès à ces con-

naissances et à les mettre en pratique, en sont privés, faute de moyens, et faute d'éducation de base. Cette situation est évidemment dommageable pour l'intérêt général mondial. Les moyens de l'améliorer relèvent évidemment d'un grand nombre de facteurs. Mais il est particulièrement intéressant d'analyser dans ce contexte l'évolution récente du droit de la propriété intellectuelle, et son rôle facilitateur ou au contraire inhibiteur du point de vue de la protection du *domaine public*⁵ et de l'intérêt général.

Comment le droit actuel de la propriété intellectuelle évolue-t-il par rapport à la défense du domaine public, comment prend-il en compte les exceptions à des fins d'intérêt général (copie privée, enseignement, recherche), tend-il à renforcer ou à affaiblir les exclusions métajuridiques, comme l'exclusion de la protection des données brutes ou des idées, procédures, méthodes, concepts mathématiques, si nécessaire pour la libre circulation des idées? Le développement de la protection intellectuelle, en faisant monter le coût de l'accès aux connaissances, favorise-t-il réellement la recherche fondamentale et le rythme des innovations? De trop fortes protections n'induisent-elles pas des rentes de situations, des monopoles juridiques sur des inventions, empêchant par là-même une plus large et plus générale diffusion du savoir et du progrès?

Comment évaluer la part du grand patrimoine commun des connaissances humaines dans toute innovation spécifique? Comment alors faire la part, en toute justice, entre ce qui revient en propre à l'innovateur et ce qui revient à l'humanité dans son ensemble?

L'évolution du droit de la propriété intellectuelle semble favoriser depuis quelques années une privatisation rampante du domaine public. Le droit de la propriété intellectuelle semble n'évoluer que pour servir toujours davantage les intérêts des plus forts, des plus puissants, par le biais de l'évolution du droit des brevets, par l'emprise croissante des normes technologiques, des standards informatiques. Cette évolution se fait sans véritable débat démocratique, et on a le sentiment qu'elle se fait au profit de groupes de pression particulièrement actifs, mobilisant les parlementaires à leur cause, tout en s'efforçant de tenir l'opinion publique éloignée de ce qui se trame.

Or la gestion des *biens communs* de l'humanité devrait désormais être traitée comme un sujet politique essentiel, touchant à la *chose publique* mondiale. L'impli-

4 François Dagognet, op.cit .

5 Le domaine public doit s'appuyer sur le droit d'auteur et non pas l'infrimer. En effet, les logiciels libres sont protégés par le droit d'auteur, et c'est parce qu'ils sont protégés par le droit d'auteur que le créateur peut imposer des conditions qui font que les utilisateurs ne peuvent les privatiser en les modifiant. L'intérêt commun suppose donc, non seulement le domaine public (qui peut être accru par un relèvement des standards de protection, un abaissement de la durée de celle-ci et des exclusions de principe – comme par exemple pour les algorithmes), mais aussi par la reconnaissance d'un droit d'accès à l'information structuré (il y a déjà une base juridique pour cela) et par l'introduction d'une exception générale de *fair use*.

cation politique principale est, nous l'avons déjà dit, le rôle de la puissance publique dans la défense de ces biens communs, qui s'ils sont laissés à eux-mêmes seront soit pillés, soit très mal répartis, soit inexistants.

Les puissances publiques peuvent décider de renforcer les droits de propriété intellectuelle (leur champ d'application, leur nature, leur durée) consentis aux inventeurs et aux créateurs pour encourager la production de connaissance. Ceux-ci disposent alors de revenus tirés de l'exploitation de leurs brevets, sur lesquels ils ont un monopole d'exploitation. Une plus grande activité inventive est encouragée, mais au dépens d'une restriction de l'utilisation des connaissances élaborées. Les connaissances sont *protégées* par un *monopole* et donc limitées dans leur usage à ceux qui peuvent payer le coût d'accès demandé par le détenteur de ce monopole. Cet effet n'est pas nécessairement du goût de tous et ne rentre pas nécessairement dans le cadre d'une politique cherchant à promouvoir le libre accès de tous à l'information ou l'utilisation la plus concurrentielle possible des informations, connaissances et inventions disponibles. La puissance publique peut alors décider de rééquilibrer cet avantage donné aux inventeurs en limitant la durée de la protection de l'invention (permettant alors un retour plus rapide dans le domaine public) ou de limiter l'étendue et la nature des inventions protégeables, en exigeant une activité inventive réellement significative.

Les connaissances techniques et scientifiques de base ainsi que les informations appartenant au domaine public sont des éléments-clé dans la production de nouvelles connaissances. Les inventeurs et les créateurs d'aujourd'hui ne sont-ils pas comme des enfants assis sur les épaules de leurs aïeux, des *nains juchés sur des épaules de géants*? Ce sont les petits-enfants du savoir et des traditions, de la richesse inventive commune amassée par la collectivité humaine dans son ensemble. Comment rétribuer l'apport effectif des nouveaux venus tout en tenant compte de l'intérêt supérieur de la collectivité? Comment éviter le hold-up toujours possible des nouveaux arrivants sur le bien commun ouvert à tous? La pratique actuelle ne reconnaît aucun droit à la collectivité humaine en tant que telle dans l'invention. L'inventeur, l'homme providentiel, le génie solitaire, le Prométhée irremplaçable est supposé incarner à lui seul l'étincelle créatrice, l'intuition salvatrice, arrachant héroïquement le feu inventif aux dieux de l'ignorance. A lui les honneurs et les brevets, à lui le monopole sur l'idée et les revenus. A lui la protection accordée par la collectivité.



Mais qui viendra défendre les droits des silencieux, les innombrables apports consentis par la collectivité pour former le savant, et surtout qui viendra parler pour défendre l'immense domaine public, dans lequel tout le monde prélève, mais que peu s'évertuent à alimenter. Comme le domaine public est à tous et que les idées ne sont à personne, il n'y a pas de prix à payer, croit-on, et chacun en particulier peut se les approprier, pourvu que la société soit assez bonne fille pour reconnaître le fait juridique de cet accaparement et accepte de le défendre par la loi et les tribunaux eux-mêmes publics. Dans la plupart des cas, l'exploitation des biens communs matériels, immatériels ou sociétaux ne donnent lieu à aucun reversement à la collectivité de la part de ceux qui ont tiré un si bel avantage, à si bon compte. Pourtant, on

pourrait défendre en toute équité et en bonne justice, mais aussi avec en vue une meilleure utilisation des ressources communes, de réclamer au déposant de brevets, un paiement proportionnel aux recettes effectives, pour dédommager la collectivité mondiale de l'usage ainsi fait du bien commun et de la protection juridique socialement reconnue, mais surtout pour permettre à cette même collectivité de préparer l'avenir, de renforcer par ce retour financier supplémentaire et à l'échelle mondiale

les écoles et les universités, les bibliothèques et les laboratoires dont les nouvelles générations d'inventeurs auront le plus grand besoin, et en particulier celles qui n'ont pas dans leurs propres pays toutes les ressources nécessaires. Bref, il faut inventer et mettre en place une solidarité inter-générationnelle et multi-latérale. Il faut prélever une part des bénéfices dégagés aujourd'hui par tous ceux qui profitent de ce bien commun qu'est le système de la propriété intellectuelle pour corriger les écarts inacceptables dans l'accès mondial à l'éducation et au savoir, et donc à l'invention. Ces écarts sont provoqués par l'absence patente d'une véritable *gouvernance mondiale* effective dans ce domaine, laissé aux mains des Etats. Pourtant, les pays les plus avancés n'ont pas eu trop de peine, parce que cela les arrangeait, à mettre au point des formes de *gouvernance mondiale* en action, permettant de creuser leur avantage relatif en matière de propriété intellectuelle. L'existence de l'Organisation Mondiale de la Propriété Intellectuelle (OMPI) témoigne d'un besoin latent d'universalité en la matière. Mais il n'est pas entièrement résolu par la création d'une telle entité. Il reste le plus important, c'est-à-dire la définition d'un équilibre politique plus général, qui relie par exemple les questions assez étroites de propriété intellectuelle aux problèmes plus vastes d'égalité dans l'accès aux savoirs, à la formation.

Il relèverait par exemple d'un discours plus véritablement *politique* de déclarer que les ressources générées par la propriété intellectuelle doivent être frappées d'un *impôt mondial*, qui serait reversé aux agences spécialisées des Nations Unies et qui servirait à augmenter les chances des pays en développement de tirer parti du bien commun mondial des connaissances, et leur permettant par là-même d'y contribuer à leur tour.

Le choix d'un système spécifique de rétribution de l'activité inventive peut avoir des conséquences extrêmement importantes sur le rythme des inventions. En protégeant de manière trop large ou trop longue, on risque tout simplement d'appauvrir les opportunités pour des inventions ultérieures, les futurs inventeurs étant d'autant plus contraints pour capitaliser sur les savoirs précédents. En protégeant de manière trop étroite, on peut tomber dans l'excès inverse et diminuer l'incitation à l'activité des inventeurs, désormais non assurés d'un retour lucratif. Quoi qu'il en soit, il est clair qu'une protection trop forte⁶ a pour conséquence d'élever le prix de l'accès aux connaissances, aux idées nouvelles, et donc accentue inévitablement le fossé entre les *info-élus*, ayants droit à l'information, et les *info-exclus*. L'existence de cette distorsion peut devenir tout à fait intolérable comme le démontre, aux Etats-Unis, pays *libéral* par excellence, l'existence de lois anti-trust visant à rétablir (contre les conséquences du *libre* fonctionnement du marché) la possibilité d'une concurrence *loyale*, fondement de l'idéologie du libéralisme économique.

Le procès mené par le gouvernement fédéral américain contre Microsoft est exemplaire à cet égard. Que les gardiens du temple du libéralisme puissent mettre en accusation l'entreprise la plus emblématique de l'économie post-industrielle n'est pas sans précédent dans l'histoire américaine. La question de la perversion du système libéral par ses meilleurs disciples s'était déjà posée en d'autres temps. L'éclatement du monopole d'ATT avait été à son époque un bon exemple du sursaut nécessaire de l'intérêt général devant la puissance acquise par les intérêts sectoriels. Mais dans le cas de Microsoft il reste à savoir si le dysfonctionnement ainsi constaté vient du fonctionnement naturel du marché, que l'on se contenterait alors de réguler par les lois anti-trust, ou alors s'il vient de l'existence d'un droit de la propriété intellectuelle mal proportionné, donnant un avantage immérité, beaucoup trop *fort*, aux détenteurs de brevets, acquérant un monopole. Autre hypothèse encore: ce dysfonctionnement ne vient-il pas de la nature même de l'économie du virtuel, à base de normes et de réseaux, et donnant ipso facto un avantage incommensurable aux premiers arrivés, ou aux premiers émergents, ou aux premiers déposants de telles ou telles composantes d'un standard technique, devenu absolument indispensable à la collectivité mondiale, et transférant alors à leurs déten-

teurs une rente de situation sans proportion avec leur activité inventive réelle?

On voit que ces questions ne sont pas seulement de nature juridique. Le droit, en l'occurrence, n'est que le champ ouvert aux chocs d'une bataille plus large, de nature profondément politique, bataille elle-même alimentée par un débat philosophique sur le concept même d'intérêt général, de bien commun, à l'échelle mondiale. Les conflits d'intérêts s'exacerbent particulièrement entre les pays en développement et les pays développés, les premiers craignant à juste titre que le renforcement du droit international de la propriété intellectuelle se fasse à leurs dépens, au moment où précisément ils auraient le plus grand besoin d'élargir leur accès aux ressources mondiales de l'information pour rattraper leur retard cognitif, scientifique et informationnel.

Pour l'observateur, il est frappant de constater que les rares forums internationaux où s'élaborent les possibilités de consensus sur ces questions de propriété intellectuelle (OMPI, OMC) sont d'abord des lieux d'exercice des rapports de force entre coalition d'intérêts sectoriels, plutôt que des lieux d'élaboration d'une doctrine philosophiquement ou politiquement acceptable par tous, visant l'intérêt général.

On a vu que les abus induits par une trop forte protection de la propriété intellectuelle peuvent être corrigés dans une certaine mesure dans les pays développés disposant de lois anti-trust. Mais de tels mécanismes n'existent pas au niveau international. L'absence de lois anti-trust mondiales, mais aussi l'absence de toute souveraineté mondiale capable de faire respecter une forme de justice mondiale en matière économique, est bien la preuve que le monde est actuellement essentiellement livré aux rapports de force.

C'est pourquoi nous pensons que le chantier de la propriété intellectuelle devrait désormais être traité, non pas seulement d'un point de vue juridique ou commercial, mais essentiellement d'un point de vue éthique et politique. ■

6 Cf. Jean Monnet, Aspects actuels de la contrefaçon, 1975: "la protection du moyen général est quelque chose de nuisible à la recherche pour la raison suivante: si l'on reconnaît que la fonction en elle-même est protégée, on aboutit à la sclérose de la recherche car le breveté, ayant une protection d'étendue énorme, sera incapable d'exploiter tout le domaine qu'il aura et de nombreux concurrents seront peu enclins à aller dans ce domaine qu'ils peuvent penser être effectivement couvert par le brevet." cité par F. Dagognet in op.cit.

LOGICIELS LIBRES ET ENTREPRISES



BERNARD LANG, INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET AUTOMATIQUE –
ASSOCIATION FRANCOPHONE DES UTILISATEURS DE LINUX ET DES LOGICIELS LIBRES – FRANCE *

EN GUISE D'INTRODUCTION

Comment un mathématicien va-t-il cuire un œuf dur, avec à sa disposition un œuf cru, une casserole vide, un robinet et un réchaud? Tout simplement en remplissant la casserole d'eau, en la faisant chauffer et en laissant l'œuf dans l'eau bouillante pendant quelques minutes. Mais tout l'intérêt de cette question réside dans un second problème où le mathématicien dispose d'un œuf cru, d'un robinet, d'un réchaud et d'une casserole d'eau bouillante. La réaction de tout bon mathématicien est instantanée: il jette l'eau bouillante, vidant ainsi la casserole pour se ramener au problème précédent. Cette histoire caricature à peine l'évolution actuelle de notre système économique. Ses règles ont été longuement élaborées et raffinées pour optimiser le développement économique et technique, dans un univers de production et de consommation contraint par la pénurie ou par les coûts de production et de distribution. Or l'économie moderne repose, pour une part toujours croissante, sur des productions immatérielles pour lesquelles ces contraintes ont largement disparu. Grâce à l'informatique et à l'Internet, les coûts marginaux de production comme les coûts de distribution sont quasiment nuls. Or, plutôt que de repenser les structures économiques et les règles qui les régissent, on constate que, pour des raisons qui semblent relever de la pure idéologie, les responsables politiques et économiques choisissent de jeter l'eau bouillante, c'est-à-dire de rematérialiser l'immatériel par des combinaisons des procédures techniques (encryptage, filigrane) et juridiques (brevets, interdiction de certains outils), afin de se ramener au problème précédent. Malheureusement, ou heureusement, cela n'est sans doute pas viable à terme. En effet, même si l'on réussit ainsi à rétablir une certaine pénurie et à maintenir, temporairement, les vieilles structures économiques, on ne peut plus stabiliser les marchés sur des coûts marginaux qui sont devenus nuls. En outre, on peut se poser la

question de savoir si, avec cette nouvelle donne de l'immatériel, ce sont toujours les vieilles recettes qui optimisent le développement économique et technique.

LES RESSOURCES LIBRES

En dehors des ressources immatérielles protégées depuis 2 à 4 siècles par le brevet ou le droit d'auteur, il existe d'autres ressources, techniquement et économiquement essentielles, mais qui restent librement utilisables, notamment celles qui constituent les connaissances scientifiques. Le monde de la connaissance scientifique est lui-même une structure économique où l'on échange des biens tels que connaissance, renommée, attention, voire des biens plus matériels, et dont le but est l'accroissement du savoir humain. Les mécanismes moteurs et régulateurs de cette économie sont fondés sur une libre circulation et réutilisation de l'information, sur le jugement des pairs, la liberté de chacun de reprendre, amender, déformer, reformuler, étendre les contributions existantes, avec pour seuls objectifs le développement de connaissances nouvelles et l'amélioration, le perfectionnement, des connaissances acquises. Le rôle de la libre diffusion est dans ce contexte tellement important que cette diffusion de la connaissance devient un objectif en soi. On ne peut que

* Cet article a été initialement présenté à la Conférence: *The influence of intellectual property on world economic development*, Monaco, 19-21 Septembre 1999.
Il a été publié dans le numéro spécial 80/81 de la revue Terminal, Éditions L'Harmattan, 2000, intitulé: *Les logiciels libres: de l'utopie au marché*.

prendre acte de ce que ce modèle a parfaitement rempli son rôle. Si bien que, dès les années 1980, il a été proposé par Richard Stallman de le reprendre pour le développement des logiciels informatiques. Sans même recourir à de récents résultats de logique qui établissent qu'un programme informatique est une entité de même nature qu'une preuve mathématique [1], on constate empiriquement que les programmes informatiques se développent dans un contexte semblable à celui de la connaissance scientifique: rôle prépondérant de la matière grise sur l'investissement, grande complexité, nécessité d'une décomposition modulaire permettant de réutiliser des résultats (programmes) en faisant confiance à leurs auteurs, techniques et résultats toujours perfectibles par les contributions et les critiques des pairs, et surtout possibilité de réutiliser à l'infini, sans coût supplémentaire, un résultat (programme) obtenu. C'est ainsi qu'une communauté d'informaticiens a depuis 20 ans entrepris de développer des ressources informatiques libres [2], les **logiciels libres**, caractérisés principalement par:

- la liberté de diffusion
- la liberté d'utilisation
- la liberté de modification .

Ce mouvement a en outre été favorisé aux États Unis par la réglementation fédérale qui impose ce statut aux produits financés par l'argent public.

Il est important de comprendre que l'expression de ces 3 libertés se décline juridiquement bien sûr, sous forme de licence de distribution, mais aussi techniquement en fournissant en particulier les ressources immatérielles nécessaires à la modification de ces logiciels: documentation et code source.

Cependant, limitée aux trois règles ci-dessus, une ressource libre n'est guère différente d'une ressource dans le domaine public. Le principal *problème* est qu'il s'agit d'une caractéristique non transmissible. Tout acteur qui effectue une modification peut prétendre à la propriété de la version modifiée, et détruire ainsi la dynamique de libre circulation et de libre contribution, et même la motivation de contributeurs éventuels. Les promoteurs privés du logiciel libre ont donc fait œuvre juridique originale en utilisant le droit d'auteur pour protéger également le libre usage des évolutions successives de leurs créations: ils ont ainsi conçu une licence de distribution, dite Licence Publique Générale (GPL) [3], qui spécifie que les 3 droits ci-dessus ne sont conférés qu'à la condition expresse que les bénéficiaires les reconduisent sur les œuvres dérivées, tant sur le plan juridique que sur le plan technique. Notamment cela impose que toute diffusion soit accompagnée des documents nécessaires à la mise en œuvre et à la modification.

On constate empiriquement 20 ans plus tard que les mêmes mécanismes, qui ont présidé à l'incontestable succès de la science occidentale, produisent effectivement

aujourd'hui - et pas seulement en milieu universitaire - des produits informatiques de grande qualité qui concurrencent avec succès ceux des plus grands acteurs du marché. Ce succès a été longuement analysé sur le plan technique, notamment dans l'essai d'Eric Raymond: **La cathédrale et le Bazar** [4] ou dans les rapports internes de la société Microsoft, dits **rapports Halloween** [5].

Mais le succès technique et la qualité ne sont pas tout. L'acceptation actuelle, voire parfois l'engouement - le logiciel Apache, par exemple, détient 56% de son marché [6], et Linux est la plate-forme serveur qui a la plus forte croissance [7] - de ces produits par les acteurs économiques impliquent la satisfaction de nombreux critères non techniques, de nature économique ou industrielle.

L'INFORMATIQUE PROPRIÉTAIRE

L'industrie du logiciel présente une grande variété de situations, due elle-même à la très grande variété des logiciels, en complexité, en rôle, en taille du produit ou du marché. Nous nous intéressons ici au secteur de l'édition de logiciels, c'est-à-dire aux logiciels à moyenne ou grande diffusion, par opposition aux logiciels très spécialisés qui relèvent plus du secteur des services que de celui de l'édition. Une caractéristique clairement observable de ce secteur est la tendance à l'instauration de monopoles, que ce soit pour les systèmes d'exploitation, les bases de données, les routeurs, les logiciels de gestion, les moteurs multimedia, voire même dans des marchés de niche comme la gestion des cabinets dentaires. Cette tendance au monopole s'explique en fait facilement par des phénomènes de rendements croissants dûs principalement à la combinaison de deux causes:

- la nullité des coûts marginaux qui permet des économies d'échelle considérables pour les acteurs dominants;

- le rôle prépondérant et massif des *externalités* de réseau [8]. Ces externalités se manifestent en amont quand les fournisseurs ou contributeurs (producteurs de composants matériels, d'applications ou de *plug-ins*) favorisent les plates-formes les plus populaires qui représentent un marché plus important. Elles sont également présentes en aval en raison du conformisme raisonné des clients autour des produits dominants ou pressentis comme tels, parce que cela paraît une garantie de pérennité – l'on confond souvent pérennité des sociétés et pérennité de leurs produits – et parce qu'il est plus facile de trouver du personnel formé ou de l'assistance technique pour les produits les plus populaires, même s'ils ne sont pas techniquement les meilleurs.

En outre, un acteur dominant a généralement le moyen de contrôler, juridiquement (par le droit d'auteur, le *copyright* ou les brevets) ou techniquement (par le secret ou le glissement temporel des spécifications [9, 10]) les protocoles de communications, les formats de données ou les interfaces hommes-machine, qui deviennent des standards de faits. Il peut ainsi contrôler l'accès au marché de ses concurrents, voire de ses partenaires.

Cette dérive monopolistique présente nombre d'inconvénients évidents:

- un moindre effort de recherche et d'innovation, voire un bridage des recherches effectuées par des tiers, dans un secteur où ce sont généralement les petites entités ou les individus qui innovent parce que l'investissement en capital est négligeable au regard de la matière grise;
- un manque de diversité *écologique* du tissu technologique, d'où une plus grande vulnérabilité aux agressions (virus et pirates par exemple) [11], qui inquiète de plus en plus les décideurs, et en particulier les gouvernements. Ce manque de diversité risque aussi de conduire à des culs-de-sac techniques, pouvant imposer des reconversions coûteuses;
- absence de pérennité des normes officielles ou des *standards* de faits, soumis au bon vouloir des acteurs dominants, d'où en particulier le risque d'un manque de pérennité des données archivées et de la mémoire des organisations;
- absence de pérennité des logiciels eux-mêmes, soumis à la stratégie discrétionnaire de l'acteur dominant;
- dépendance des clients et de nombreux autres partenaires économiques, tant pour les prix [12] que pour la satisfaction de besoins spécifiques. Cet aspect est particulièrement critique si l'on se rappelle que l'informatique est le système nerveux de l'entreprise [13];
- dépendance du client pour sa stratégie industrielle si les logiciels sont intégrés à des produits, sans alternative en cas de conflit avec le fournisseur;

- difficulté de sécuriser un système dont on ne connaît pas les spécifications internes, et risque avéré de diffusion de code piégé [14, 15].

Nombre des problèmes évoqués sont liés au secret de la fabrication des logiciels, à la non diffusion des codes sources et de la documentation, assortis généralement de l'interdiction légale de procéder à un *retro-engineering* (désassemblage ou décompilation), ce qui serait de toutes façons extrêmement coûteux. Même dans les situations qui restent concurrentielles, le modèle commercial propriétaire, sans diffusion des sources (ou même souvent avec disponibilité contrôlée des sources) continue à présenter plusieurs des inconvénients cités, auxquels viennent s'ajouter:

- une absence de pérennité liée non seulement aux décisions des entreprises, mais aussi aux avatars de leur existence. La non-disponibilité des données nécessaires à l'entretien des logiciels peut se révéler catastrophique si l'on sait que le fournisseur peut parfois disparaître, ou perdre ces informations. Le bug de l'an 2000 en a fourni des exemples;
- une absence fréquente de standardisation effective due à la recherche de marchés captifs, ou même simplement à l'évolution divergente de logiciels initialement semblables (UNIX par exemple), dans un contexte où chaque acteur se doit de procéder à des développements indépendants. Remarquons aussi que le secteur de l'édition de logiciel est fort peu créateur d'emplois, notamment au regard de ses profits et de sa capitalisation boursière [16], à la différence évidente du secteur des services informatiques.

LES LOGICIELS LIBRES ET L'ENTREPRISE

S'ils sont clairement issus du monde universitaire et de son éthique de la connaissance, les logiciels libres ont très ra-

pidement pénétré le monde de l'entreprise, en raison de leurs qualités techniques – dont l'analyse sort du cadre de cette présentation – et de leur flexibilité d'emploi. Ainsi, dès la fin des années 1980, nombre d'outils logiciels libres étaient utilisés sur les plates-formes de type Unix (traitement de documents, environnements de programmation, compilateurs). Les promoteurs de ces logiciels, conscients de l'importance des synergies avec le monde de l'entreprise, avaient d'ailleurs fait évoluer la GPL en créant la LGPL [17], Licence Publique Générale pour les Bibliothèques, de façon à mieux encadrer la notion de logiciel dérivé, et permettre ainsi aux entreprises de créer des produits propriétaires compatibles avec un contexte de produits libres.

Les logiciels libres, par nature, échappent aux problèmes évoqués dans la section précédente. Il n'est bien entendu plus question de monopole de la production puisque tout un chacun est libre de faire évoluer et de redistribuer le logiciel. Contrairement à ce que l'on pourrait craindre, l'expérience montre que cette liberté n'entraîne pas de divergence technique significative. Cela est sans doute dû à ce que les meilleures solutions à chaque problème sont librement réutilisables par tous et s'imposent donc de façon uniforme, et aussi à la pression des communautés d'utilisateurs qui ne se fient généralement qu'à un petit nombre de concepteurs ou de distributeurs dont la réputation est bien établie, comme le montre l'exemple du système d'exploitation Linux. De plus l'existence de cette concurrence dans le développement de chaque aspect d'un logiciel, sanctionnée exclusivement par la qualité des résultats, est l'un des facteurs qui assurent l'excellent niveau technique des logiciels libres.

Toute organisation ou personne utilisant des logiciels libres a la maîtrise complète de ses outils, de leur mise en œuvre ou de leur adaptation à des besoins spécifiques, soit en assurant elle-même les tâches à accomplir, soit en les confiant à un prestataire de service de son choix. L'un des effets est donc de remplacer une industrie d'édition, peu créatrice d'emplois et fournissant une ressource peu flexible, par des activités de services permettant une meilleure adaptation des ressources techniques aux besoins spécifiques. Pour les besoins courants, la maintenance et l'adaptation des logiciels libres sont assurées par la communauté des développeurs qui ne sont généralement qu'un sous-ensemble actif des utilisateurs universitaires ou industriels.

C'est aussi l'existence de cette communauté ouverte et son examen permanent des codes sources qui y garantit l'absence de pièges et chevaux de Troie.

Le faible coût des solutions informatiques libres permet aux petites PME d'accéder plus facilement à l'informatisation de leur activité et de devenir ainsi plus compétitives. Et ces entreprises étant de loin les plus nombreuses, cela élargit d'autant le marché pour les prestataires de services informatiques.

En ce qui concerne les entreprises, de plus en plus nombreuses, qui incorporent des logiciels dans leurs produits, l'utilisation de ressources libres leur permet de maîtriser complètement la source de leur matière première, et donc leur stratégie industrielle. En outre, sa gratuité effective - même si elle ne fait pas partie des axiomes fondateurs - permet d'abaisser substantiellement les coûts et donc d'améliorer la compétitivité. C'est d'ailleurs, avec les serveurs d'entreprise et les serveurs du Web, l'un des secteurs où la croissance industrielle de l'usage des logiciels libres est la plus forte [18], pour les raisons exposées ci-dessus, et aussi parce que dans ce contexte les externalités de réseau sont les plus faibles.

LA CRÉATION DE LOGICIELS LIBRES

Une critique fréquemment formulée au sujet de ce modèle de développement concerne sa viabilité économique. Il semble peu crédible que la bonne volonté des universitaires et de quelques ingénieurs suffise à satisfaire la diversité des besoins du monde industriel. C'est peut-être vrai – encore que la réalisation du système Linux est impressionnante –, mais la création de logiciels libres s'est maintenant largement étendue aux entreprises et à d'autres organisations. Il est probablement prématuré d'analyser ce phénomène en pleine mutation, dont les succès ou les échecs sont pour la plupart encore à venir, mais on peut déjà identifier un certain nombre de mécanismes et de motivations qui favorisent cette évolution.

Depuis une dizaine d'années on voit apparaître des entreprises explorant des modèles de développement divers combinant la création et le service pour les logiciels libres, comme Cygnus, Ada Core Technologies, ou Aladdin, et dont quelques unes ont une croissance honorable. La liberté du logiciel est sou-

vent pour elles un moyen de mieux diffuser un produit pour lequel elles ont une expertise reconnue, et monnayable de diverses façons, comme le développement à façon ou la vente de nouvelles versions temporairement non libres de leurs produits. Une autre source de logiciels libres provient d'entreprises ou d'organisations qui développent des logiciels pour leurs besoins propres, sans qu'ils aient un rôle important dans leurs activités compétitives. Ces organisations ont alors intérêt à mettre ces logiciels à la disposition de tous pour bénéficier des améliorations qui pourront y être apportées par d'autres utilisateurs. Il s'agit en fait tout simplement d'amorcer un mécanisme de mutualisation des coûts de développement d'une ressource utile. Le partage d'un même fournisseur par plusieurs clients n'est souvent rien d'autre qu'une telle mutualisation des coûts. Cependant, compte tenu des risques de dérive monopolistique, on peut penser que, surtout pour des ressources critiques, la mutualisation des coûts peut être mieux assurée par le financement en commun de ressources libres, en jouant dans le cas du logiciel sur l'absence de coûts marginaux. C'est très précisément le raisonnement d'un consortium de douze grandes multinationales impliquées dans la réalisation de produits contenant des logiciels embarqués, et qui a fait réaliser par la société Cygnus le système d'exploitation libre eCOS destiné à promouvoir un standard commun [19]. La volonté de stabiliser et de pérenniser des standards est d'ailleurs un grand pourvoyeur de logiciels libres. Le ton fut initialement donné par l'IETF (*Internet Engineering Task Force*) qui structura et dynamisa le développement de l'Internet par la spécification de protocoles standardisés, dont la viabilité et la diffusion étaient assurées par l'existence d'implémentations libres. Le consortium W3C qui préside au développement du Web continue à encourager cette approche. Mais on peut aussi classer dans la même catégorie:

- la *libération* du navigateur de Netscape, essentiel pour éviter la mainmise de Microsoft sur les protocoles du Web, et de là sur le marché des serveurs puis des services en ligne;
- la *libération* du compilateur Jikes d'IBM pour le langage Java qui en se diffusant peut garantir l'indépendance de ce langage dont le rôle devient prépondérant dans les systèmes d'information;
- les interprètes libres du format MP3, destinés à assurer la survie de ce format audio contre la volonté de quelques grands éditeurs de musique.

Les logiciels libres ont montré que la présence d'une large communauté d'utilisateurs/développeurs est le meilleur garant de pérennité qui soit. Ce message a maintenant été reçu par un certain nombre de grandes entreprises qui veulent utiliser ce modèle, ou l'on déjà fait, pour pérenniser des lo-

giciels qui ne s'inscrivent plus dans une stratégie commerciale d'éditeur mais qu'elles souhaitent néanmoins préserver [20], par exemple en raison de l'existence d'une base installée ou pour maintenir une activité de service. Il ne faut cependant pas croire que la simple *libération* est une recette miracle. Il est impératif qu'une telle entreprise soit structurée par l'équivalent d'un *business plan* destiné à susciter la création d'une communauté susceptible d'assurer la pérennité recherchée.



L'AVENIR?

En dépit d'une idéologie ambiante résolument hostile a priori, le développement des logiciels libres montre que ce modèle ancien et efficace de création de richesses peut jouer un rôle essentiel dans certains secteurs immatériels de l'économie moderne. En s'opposant aux dérives monopolistiques et au secret industriel qui rigidifient les marchés et stérilisent le développement technologique, ces ressources libres sont un ferment de la croissance. Citons la revue *Stratégie Télécoms & Multimédia* (27/08/99):

*«L'envolée des logiciels libres s'est étendue récemment au marché des logiciels du domaine des télécommunications par la création de **Open Source for Open Telecom**. Les exploitants ont été contraints de s'intéresser à cette nouvelle démarche qui dynamise le marché et augmente les échanges à moindre coût.»*

Il est cependant évident que cette situation ne fait pas nécessairement le bonheur de tous les acteurs, et que les avantages économiques et technologiques de cette libéralisation ne sont pas encore bien perçus comme l'une des données essentielles du développement de l'économie moderne. On ne peut donc que s'inquiéter de voir se renforcer les *lobbies* et les législations [21] qui veulent élever les barrières protectionnistes des brevets en matière de logiciels informatiques, que ce soit pour des rai-

sons idéologiques, par ignorance des enjeux, pour défendre des intérêts corporatistes, ou par volonté clairement exprimée de bloquer le développement des logiciels libres [22, 23, 5]. Les brevets ont été institués pour dynamiser le développement des techniques, en favorisant leur découverte, leur dissémination ou leur exploitation dans l'intérêt général. Ils consistent en l'octroi par la puissance publique d'un monopole destiné à motiver ces processus et/ou à en compenser financièrement les coûts, monopole qui est lui-même un coût imposé à la société. Leur efficacité et leur utilité sont donc très dépendantes des motivations des acteurs et des conditions sociologiques et économiques dans lesquelles ces trois processus se réalisent, et notamment la nature des investissements et des infrastructures nécessaires (qui sont généralement très faibles dans le cas des logiciels) [24]. Il est donc extrêmement imprudent d'aller contre le principe reconnu depuis toujours de non-brevetabilité des créations immatérielles [25], sans même avoir analysé longuement – et cela n'a guère été fait – les conséquences durables d'une telle évolution, notamment en ce qui concerne les équilibres du marché.

RÉFÉRENCES ET NOTES

- [1] **W.A. Howard**, The formulae-as-types notion of construction, in *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, J.P. Seldin et J.R. Hindley, Academic Press, 1980.
- [2] Le terme *libre* correspond en anglais à *open source*, <http://www.opensource.org/> ou à *free*, <http://www.fsf.org/>, au sens de *libre* et non au sens de «gratuit». Le choix entre ces deux termes en anglais relève plus de questions de linguistique ou de marketing, que d'une différence technique significative. L'antonyme de *libre* est *propriétaire* utilisé comme qualificatif. Les deux types de logiciels peuvent être commercialisés.
- [3] **Free Software Foundation (FSF)**, GNU General Public License (GPL), Version 2, juin 1991, <http://www.fsf.org/copyleft/gpl.html>
- [4] **Eric S. Raymond**, *The Cathedral and the Bazaar*, First Monday, Vol.3 No.3, mars 1998, http://www.firstmonday.org/issues/issue3_3/raymond/index.html – Traduction française de Sébastien Blondeel: http://www.linux-france.org/article/these/cathedrale-bazar/cathedrale-bazar_monoblock.html
- [5] **Vinod Valloppillil**, *The Halloween Documents* – Rapports internes, Microsoft Corp., Août 1998, <http://www.opensource.org/halloween>
- [6] Netcraft: «The Netcraft Web Server Survey», <http://www.netcraft.co.uk/survey/>
- [7] **Stephen Shankland**, *Linux shipments up 212 percent*, CNET News, 16 décembre 1998, <http://www.news.com/News/Item/0,4,30027,00.html>
- [8] **Nathan Newman**, *From Microsoft Word to Microsoft World: How Microsoft is Building a Global Monopoly*, NetAction, 1997, <http://www.netaction.org/msoft/world/>
- [9] **Robert X. Cringely**, *Why the Right Price for Your Next PC is Nothing*, The Pulpit, 3 septembre 1998, <http://www.pbs.org/cringely/pulpit/pulpit19980924.html>
- [10] **Roger Irwin**, *Shifting Standards*, November 1998, <http://www.geocities.com/SiliconValley/Hills/9267/sstactics.html>
- [11] **Jamais Cascio**, *The ecology of computer viruses*, Salon Magazine, 7 avril 1999, <http://www.salonmagazine.com/tech/feature/1999/04/07/melissa/print.html>
- [12] **Jesse Berst**, *Windows pricing could be the smoking gun*, ZDNet AnchorDesk, 13 avril 1998, <http://www.zdnet.com/pcweek/opinion/0413/13berst.html>
- [13] **Microsoft**, *Bill Gates Keynotes Worldwide Developer Conference*, Microsoft PressPass, 2 septembre 1998, <http://www.microsoft.com/presspass/press/1998/sept98/devancpr.htm>
- [14] **Andrew Fernandes**, *Microsoft, the NSA, and You*, Press Release, Cryptonym Corporation, 3 septembre 1999, <http://www.cryptonym.com/hottopic/msft-nsa.html>
- [15] **Duncan Campbell**, *NSA Builds Security Access Into Windows*, CMP's TechWeb, 9 mars 1999, <http://www.techweb.com/wire/story/TWB19990903S0014>
- [16] **Microsoft Corporation**, *Microsoft Fast Facts*, juin 1999, <http://www.microsoft.com/presspass/fastfacts.htm>
- [17] **Free Software Foundation (FSF)**, *GNU Library General Public License (LGPL)*, Version 2, juin 1991, <http://www.fsf.org/copyleft/lgpl.html>
- [18] **RHI Consulting**, *Survey says future looks bright for LINUX technology*, Press release, 28 septembre 1999, http://www.rhicconsulting.com/jobsRHIC/About_RHIC/092899.html
- [19] **Cygnus Solutions**, *eCos - embedded Cygnus operating system*, 1999, <http://www.cygnus.com/ecos/>
- [20] **Bull**, *Bull supports open source initiative with source code for Enterprise JavaBeans (EJB) platform*, Communiqué de presse, 24 juin 1999, http://www.openmaster.com/press/releases/pr_990624_ejb.htm
- [21] **UNDP, United Nations Development Program**, *Human Development Report 1999*, Oxford University Press, 1999, <http://www.undp.org/hdro/index2.html> – The relentless march of intellectual property rights needs to be stopped and questioned
- [22] **La Commission européenne**, *Brevets: la Commission expose les grandes lignes d'un plan de mesures ambitieux*, 12 février 1999, <http://europa.eu.int/comm/dg15/fr/intprop/indprop/99.htm>
- [23] **Didier Lombard**, *Le Brevet pour l'Innovation*, Synthèse du rapport Lombard, mars 1998, <http://www.evariste.anvar.fr/inpi/pi980121.html>
- [24] **Jean-Paul Smets**, *Software Useright: Solving Inconsistencies of Software Patents*, septembre 1999, <http://www.smets.com/it/policy/useright/>
- [25] *Convention sur la délivrance de brevets européens*, article 52, 5 octobre 1973, <http://www.european-patent-office.org/epc97/epc97-f.htm#ar52>
- [26] **Jean-Paul Smets et Benoît Faucon**, *Logiciels libres - Liberté, égalité, business*, Edispher, Paris 1999, <http://www.freepatents.org/liberty/> ■

Libre reproduction © Copyright Bernard Lang

FSF Europe

LIBERTÉ ET COOPÉRATION

FRÉDÉRIC COUCHET, ASSOCIATION FSF EUROPE – CHAPITRE FRANCE,
FREDERICCOUCHET@FSFEUROPE.ORG



RÉSUMÉ

Ce document est un article de présentation de la *Free Software Foundation Europe* (FSF Europe) et plus particulièrement de la FSF France.

PRÉSENTATION

La FSF Europe est l'organisation européenne sœur de la *Free Software Foundation* fondée en 1985 par Richard M. Stallman (l'organisation américaine historique dédiée à l'abolition des restrictions sur l'utilisation, la compréhension, la redistribution et la modification des programmes informatiques). La FSF promeut le développement et l'utilisation du Logiciel Libre, et est à l'origine du projet GNU, qui a notamment donné naissance au système d'exploitation GNU/Linux.

Comme la FSF originelle, la FSF Europe fournit une organisation politique et lobbyiste pour le Logiciel Libre et le projet GNU. Elle travaille pour devenir l'épine dorsale pour les parties du projet GNU basées en Europe et entretient des contacts étroits avec la presse et les représentants politiques pour accroître la couverture du Logiciel Libre et du projet GNU. Elle aide en outre à lever et à distribuer des fonds pour le Logiciel Libre partout en Europe. Pour cette raison, elle est exempte de taxes partout où c'est possible.

ORGANISATION

Comme il n'y a pas de législation européenne commune pour les institutions, la FSF Europe est organisée de façon modulaire. Elle consiste en une organisation pivot fondée à Hambourg (Allemagne) avec des sous-organisations dans chaque pays d'Europe. Ces chapitres locaux ont un chancelier (président) ainsi qu'un vice-chancelier (secrétaire). Ils sont disponibles pour la presse locale et les représentants politiques, et sont en contact permanent avec les communautés du Logiciel Libre du pays. Des organisations, auxquelles tout le monde peut prendre part, pourront devenir organisations associées et constituer les muscles du squelette de la FSF Europe. Enfin, les volontaires pourront participer notamment via Savannah (<http://savannah.gnu.org>) – voir plus loin. De cette façon, chacun peut être impliqué dans le Mouvement du Logiciel Libre et la FSF Europe.

La FSF Europe travaille de concert avec ses organisations associées. Deux de ces organisations sont enregistrées en France: l'Association pour la Promotion et la Recherche en Informatique Libre (APRIL, fondée en 1996) qui fut la première organisation associée à la FSF Europe, et l'*Organization for Free Software in Education and Teaching* (OFSET), très impliquée dans le domaine de l'éducation.

Objectifs

La FSF Europe a deux objectifs principaux:

Compléter l'offre de Logiciel Libre

Aujourd'hui il n'existe pas de Logiciels Libres pour tous les besoins. La FSF, tout comme la FSF Europe, œuvre à compléter l'offre Logiciel Libre afin qu'il soit possible d'utiliser un Logiciel Libre quel que soit le domaine d'activité. Par exemple, il est aujourd'hui difficile voire impossible de transmettre efficacement du son et de l'image sur les réseaux en utilisant du Logiciel Libre.

Le rôle de la FSF et de la FSF Europe est d'encourager les entreprises et les individus à identifier les friches du Logiciel Libre et à les compléter. Il suffit souvent d'un simple dialogue pour qu'un créateur réalise qu'il peut combler un manque, compléter une brèche dans l'univers Logiciel Libre. Ainsi, à l'occasion des FSF Awards au début de l'année 2001, après un dialogue intense et constructif, l'Ecole Centrale Paris (ECP) a édité sous licence GNU GPL les développements de VideoLAN, ajoutant ainsi une pierre importante à l'édifice.

Par ailleurs, la FSF étant à l'origine de l'univers Logiciel Libre qui existe aujourd'hui, les entreprises pourront s'y intégrer facilement par son intermédiaire. Un aspect que réalisent difficilement les individus et les entrepreneurs est la facilité d'établir des relations bilatérales dans la communauté du Logiciel Libre. Une entreprise développant son activité sur MySQL, GNU/Linux et Perl peut facilement entrer en contact avec les groupes d'individus et entreprises qui développent et maintiennent ces éléments. Ils n'ont besoin de personne pour établir ce contact mais l'expérience nous a montré qu'une réticence naturelle empêche qu'il s'établisse. Le rôle de la FSF Europe est alors simplement de montrer comment établir ce contact et maintenir un dialogue. L'éloignement, la communication par e-mail, la sociologie des groupes de développement rendent cet exercice suffisamment inhabituel pour qu'une aide extérieure soient la bienvenue.

En bref, établir des relations avec la communauté du Logiciel Libre est un aspect indispensable qui doit compléter l'activité de rapprochement avec les entreprises au niveau national. Les entreprises doivent pouvoir s'intégrer au mouvement du Logiciel Libre et non pas uniquement se reposer dessus. Elles consolideront ainsi leurs fondations. Aussi impressionnant qu'il y paraisse, le mouvement du Logiciel Libre est assez petit pour qu'un groupe de quelques entreprises puissent y jouer un rôle considérable; c'est une chance à saisir.

Combattre les dangers qui menacent les fondements du Logiciel Libre

Les lois autour du logiciel sont très mouvantes car il s'agit d'une science jeune. Par manque d'information, nos législateurs peuvent être amenés à prendre des mesures qui interdisent l'existence du Logiciel Libre. Les quatre libertés fondamentales du Logiciel Libre sont encore fragiles:

UTILISATION

la liberté d'utiliser le logiciel, pour quelque usage que ce soit;

ÉTUDE

la liberté d'étudier le fonctionnement du programme, et de l'adapter à vos propres besoins. L'accès au code source en est une condition;

REDISTRIBUTION

la liberté de redistribuer des copies de façon à pouvoir aider votre voisin;

MODIFICATION

la liberté d'améliorer le programme, et de diffuser vos améliorations au public, de façon à ce que l'ensemble de la communauté en tire avantage. L'accès au code source en est également une condition.

Par exemple, la loi SIAE en Italie impose que chaque logiciel destiné à être vendu fasse l'objet d'un enregistrement officiel. Cet enregistrement a bien entendu un coût. Ainsi, vendre une distribution GNU/Linux (Debian par exemple) impose d'enregistrer individuellement les milliers de logiciels qu'elle contient. Cette nouvelle loi handicape sérieusement la liberté de distribution et menace donc directement l'existence du Logiciel Libre.

Une autre menace est constituée par les brevets logiciels. Chaque Logiciel Libre est potentiellement à la merci d'un brevet logiciel. En effet il est pratiquement impossible de déterminer avec certitude si un logiciel tombe sous le coup d'un brevet ou non. Si les brevets logiciels devenaient légaux en Europe il serait donc potentiellement illégal d'utiliser, modifier ou redistribuer l'ensemble des Logiciels Libres.

Nous souhaiterions que la situation soit moins périlleuse. Mais comme ce n'est pas le cas, la FSF Europe doit lutter pour consolider les fondements légaux de l'univers des Logiciels Libres. La tâche est d'autant plus importante que la FSF est auteur de la licence GNU GPL qui couvre près de 70% des Logiciels Libres existants. La FSF Europe peut fournir aux entreprises et aux administrations une information précise sur certains problèmes légaux. Elles pourront agir et veiller avec nous pour consolider les bases juridiques du Logiciel Libre. La FSF Europe s'engage actuellement à assister et coordonner le processus d'adaptation de la licence GNU GPL aux droits français et allemand. De manière plus générale, elle met en place un réseau de juristes compétents dans le domaine du Logiciel Libre.

La FSF Europe joue un rôle fédérateur dans chaque pays, rassemblant les acteurs du Logiciel Libre afin de leur permettre de communiquer. Grâce aux présences que nous avons établies en Suède, France, Allemagne, Italie et bientôt en Suisse, au Portugal, en Belgique et en Autriche nous pouvons agir concrètement au niveau européen. Dans chaque pays des relations de travail se mettent en place avec les associations qui défendent depuis de nombreuses années le Logiciel Libre afin d'apporter une dimension européenne à leur action. L'APRIL en France peut par exemple coordonner les efforts de traduction de la licence GNU GPL avec des juristes français.

ACTUALITÉ

La FSF Europe a démarré officiellement ses activités le 10 mars 2001. Sa première assemblée générale a eu lieu le 6 mai 2001, à Essen (Allemagne). Les premiers chapitres locaux (Allemagne et France) sont actuellement créés et actifs. Les chapitres suédois et italien devraient être actifs très bientôt.

Chronologie des actions du chapitre français

La FSF Europe a, de fait, commencé ses activités en France le 12 avril 2001. Elle a débuté par les aspects administratifs (par exemple l'hébergement du siège social par Lolix) et d'infrastructure (notamment l'installation de la machine fr.fsf.org, première machine intégrée au projet GNU présente hors des USA, chez Nevra). L'activité quotidienne du chapitre français utilise des ressources GNU (DNS, Savannah). Une des premières tâches importantes a été l'établissement de relations de travail avec la communauté française du Logiciel Libre (notamment à travers l'APRIL et OFSET).

Traduction de la GNU GPL en français

Mélanie Clément Fontaine, juriste chez Alcôve et doctorante en propriété intellectuelle, a démarré la traduction et l'adaptation au droit français de la GNU GPL. Elle reçoit maintenant le support de la FSF Europe, qui démarre également un effort similaire en Allemagne. Plus généralement, la FSF Europe va regrouper des juristes compétents et intéressés par le Logiciel Libre pour constituer un réseau d'experts qui pourra l'aider dans le futur.

Rencontres

La FSF France a rencontré de nombreuses personnes à l'occasion de diverses manifestations (comme **Libre et Vie Locale** par exemple) et a commencé à établir des contacts avec différents projets et institutions. Le premier exemple concret fut la rencontre avec les responsables de la future Technopole Logiciel Libre à Soissons. Soutenue par le gouvernement français et en liaison étroite avec l'université de Soissons, la Technopole Logiciel Libre accompagnera les entreprises développant une activité de services basée sur les Logiciels Libres. En dispensant des formations initiant les chefs d'entreprise aux modèles économiques créés par le Logiciel Libre, la technopole redonnera vie à un paysage économique appauvri par la disparition des

industries les plus importantes de la région. La dernière en date concerne une réunion avec la MTIC (Mission interministérielle de soutien technique pour le développement des Technologies de l'Information et de la Communication dans l'administration) et les responsables du **bouquet du libre**. Ces rencontres ont été l'occasion de présenter la FSF France et aussi de mettre en relation différentes personnes intéressées par des sujets connexes, mais qui ne se connaissaient pas.

Savannah et l'Europe

Savannah (<http://savannah.gnu.org/>) est un point central pour le développement, la distribution, la maintenance de logiciels libres. Savannah met à disposition, sous la forme d'un site Web, un ensemble d'outils tels qu'un environnement pour archivage, un système de gestion de versions, un gestionnaire de tâches, des listes de diffusion, un espace FTP.

Savannah a été mis en place par le projet GNU, mais est utilisable par tout projet Logiciel Libre. Grâce aux dons de membres de l'APRIL et de la FSF Europe, une nouvelle machine a été installée pour Savannah. Les développeurs les plus actifs de Savannah sont en France et au Portugal. Nous avons également pris contact avec les responsables de deux projets similaires à Savannah (PeCoVall, Pure/Source) et discuté de la manière de partager les efforts et les ressources pour obtenir une plateforme de développement coopérative encore plus puissante.

Sensibilisation

La FSF Europe a eu des entretiens avec des journalistes pour présenter ses objectifs et ses activités. La présence lors de diverses manifestations a permis également de sensibiliser les gens sur l'importance de la FSF Europe.

Où puis-je obtenir d'autres informations?

Pour obtenir davantage d'informations sur la FSF Europe, vous devriez vous abonner aux listes de diffusion. Pour ce faire et pour accéder aux archives, rendez-vous sur la page d'accueil: <http://www.fsfeurope.org>

Vous pouvez également entrer en contact avec les personnes mentionnées ci-dessous ou envoyer un courrier électronique à toute l'équipe de la FSF Europe à team@fsfeurope.org. Vous pouvez également prendre des nouvelles de la FSF France à l'adresse <http://france.fsfeurope.org/news/>.

RÉFÉRENCES

- [1] Free Software Foundation - <http://www.fsf.org/>
- [2] Projet GNU <http://www.gnu.org/>
- [3] Free Software Foundation Europe <http://www.fsfeurope.org/>
- [4] Free Software Foundation Europe-Chapter France alias FSF France <http://france.fsfeurope.org/>
- [5] Savannah <http://savannah.gnu.org/projects/fsfe/>
- [6] Source Forge <http://SourceForge.net/>
- [7] APRIL <http://www.april.org/>
- [8] OFSET <http://www.ofset.org/>
- [9] VideoLan <http://www.videolan.org/>
- [10] École Centrale de Paris <http://www.ecp.fr/>
- [11] Loi SIAE <http://www.fsfeurope.org/siae/siae.fr.html>
- [12] Lolix <http://www.lolix.org/>
- [13] NevraX <http://www.nevraX.org/>
- [14] Alcôve <http://www.alcove.com/>
- [15] Technopole Logiciel Libre <http://www.soissons-technopole.org>
- [16] MTIC <http://www.mtic.pm.gouv.fr/>
- [17] PeCoVall <http://pecovall.eu.org/> ■

Copyright © 2001 FSF Europe – Chapter France [la reproduction exacte et la distribution intégrale de cet article sont permises sur n'importe quel support d'archivage, pourvu que cette notice soit préservée].

LES LOGICIELS LIBRES DANS L'ÉDUCATION NATIONALE FRANÇAISE



JEAN-PIERRE ARCHAMBAULT,
CENTRE NATIONAL DE DOCUMENTATION PÉDAGOGIQUE EN FRANCE (CNDP) –
MISSION VEILLE TECHNOLOGIQUE, jpierre.archambault@poste.cndp.fr *

En France, les logiciels libres suscitent un intérêt croissant dans l'Éducation nationale, de par des enjeux communs avec ceux d'autres secteurs d'activité, mais aussi des enjeux spécifiques au monde de la pédagogie. Ils rencontrent des préoccupations fortes et récurrentes du système éducatif en matière de Technologies de l'information et de la communication. Ils sont donc à connaître à la fois pour eux-mêmes et pour quelques problématiques qu'ils contribuent à poser, de la libre diffusion à tous de la connaissance à la marchandisation de certaines activités éducatives en passant par la propriété intellectuelle et l'économie des biens informationnels.

Un logiciel libre n'est pas nécessairement gratuit, même si l'on peut se le procurer dans la grande majorité des cas à des prix très nettement inférieurs à ceux des logiciels commerciaux classiques. La confusion libre/gratuit vient de l'expression *free software*, le terme *free* signifiant indifféremment en anglais libre ou gratuit. On peut copier et diffuser un logiciel libre en autant d'exemplaires que l'on veut, sans avoir à payer le moindre centime supplémentaire. On voit immédiatement tout l'intérêt que cela présente pour un établissement scolaire, précisément pour sa ligne budgétaire dédiée aux logiciels. On peut en effet se demander s'il est toujours très pertinent de dépenser des sommes respectables pour doter en nombre les établissements scolaires d'outils bureautiques, alors qu'existent des solutions équivalentes à moindre prix du côté des logiciels libres, ou de Lotus d'ailleurs. S'ouvrent également des perspectives pour les élèves et les enseignants de retrouver le même environnement de travail à la maison, sans être confrontés à des problèmes juridiques et financiers. Un logiciel libre est donné avec son code source [1], c'est à dire son secret de fabrication, contrairement à un logiciel propriétaire dont l'éditeur ne fournit que le code objet, l'*executable*. On a ainsi la possibilité d'étudier comment il fonctionne, de le modifier pour l'adapter à ses propres besoins.

Enfin, toutes ces libertés ne sont accordées qu'à la condition que les bénéficiaires les reconduisent sur les œuvres dérivées. Des licences, dont la plus connue est la GPL (General Public License), constituent un cadre juridique contractuel à cette approche.

ÉDUCATION NATIONALE: LE CADRE

En octobre 1998, le ministère de l'Éducation nationale signe avec l'AFUL (Association francophone des utilisateurs de Linux et des logiciels libres) un accord-cadre indiquant que les logiciels libres en général, Linux en particulier, constituent une solution alternative pour les établissements scolaires, dans une perspective de pluralisme technologique.

Lors du Salon de l'Éducation de novembre 1999, le ministère distribue un texte *Les logiciels libres et l'éducation* dans lequel on peut lire: «*Les logiciels libres et le système d'exploitation Linux ouvrent des perspectives extrêmement intéressantes dans le cadre du déploiement des technologies d'information et de communication dans le monde de l'éducation... Ils peuvent contribuer de manière pertinente à l'objectif de généralisation des usages de ces technologies en proposant, à très faible coût, des solutions alternatives bien adaptées à la diversité des situations qu'on rencontre dans le système éducatif...*».

Le conseil interministériel consacré à la société de l'information, qui se tient le 10 juillet 2000, retient le fait que *les logiciels libres et ouverts, et en particulier Linux, offrent des perspectives intéressantes dans le développement actuel des technologies de l'information.*

Le Bulletin officiel (BO) spécial d'août 2000 sur le *dispositif de soutien à la production multimédia* stipule que l'un des critères techniques de choix est le fait que «*les produits fonctionnent sous les trois systèmes d'exploitation les plus répandus (Windows, MacOS et Linux)*».

En janvier 2001, le ministre de la Fonction publique déclare «*Enfin, pourquoi ne pas le dire, le développement coopératif, qui est le propre des logiciels libres, la transparence et la mutualisation, qui sont à la base de leur création, sont des valeurs que nous partageons*».

Le même mois, dans une interview au journal *Le Monde*, le député Thierry Carcenac, chargé de présider une mission par le premier ministre, déclare: «*Nous ne souhaitons pas aller vers des préconisations impératives et qui ficelleraient l'administration. Nous voulons l'amener, dans le cadre de ses schémas informatiques, à ouvrir une réflexion sur le logiciel libre et à faire ses choix*».

* Article paru dans *Direction*, magazine du syndicat national des personnels de direction de l'Éducation nationale française (SNPDEN)

L'EXISTANT ÉDUCATIF

Dans ce contexte, la Mission veille technologique du CNDP mène une action d'information, selon des modalités diversifiées [2], avec l'objectif que décideurs, chefs d'établissements et enseignants se fassent leur opinion et opèrent leurs choix ultérieurs en connaissance de cause.

Des universités, des académies, des départements et des établissements scolaires, notamment les *gros* lycées, mettent en œuvre des solutions Linux: serveurs de fichiers, d'imprimantes, d'accès Internet (ou Intranet), de filtrage (pare-feu), proxy, Web, FTP, messagerie, forums.

Ainsi l'académie de Grenoble déploie-t-elle son architecture SLIS (serveur de communication Linux pour l'Internet scolaire) pour la mise en réseau et l'accès à Internet des lycées, collèges et écoles. Des fonctions telles que serveur de messagerie ou serveur Web sont rajoutées au réseau local sans le perturber. SLIS intervient aussi lors du fonctionnement au quotidien (une connexion automatique par semaine pour voir s'il y a des mises à jour; si un problème est non

des machines de récupération, des 486 ou Pentium de première génération, reliées à un serveur de type Pentium. Une fois la configuration installée, la fiabilité du système minimise les contraintes de l'enseignant.

A fonctionnalités similaires, on reconnaît aux logiciels libres des points forts dans les registres suivants: licences, adaptation, productivité, fiabilité, correction rapide et nombre limité de bogues. Par contre, le déploiement, *l'auto-maintenance* de son PC restent un problème à ce jour. Cependant, des distributions [3] sont proposées, qui permettent de configurer les machines en double amorçage et de choisir au démarrage son système d'exploitation (Linux ou Windows par exemple). Dans l'Education nationale, il faut constituer un potentiel de compétences analogue à celui construit en son temps pour les systèmes de Microsoft et de Novell. Des risques existent d'une segmentation du monde Linux si les éditeurs se mettent à proposer des solutions pour une part incompatibles. Pour le poste de travail individuel l'usage de la bureautique est encore peu répandu. Le tissu des sociétés de services doit se densifier.



DE GAUCHE À DROITE: ROBERTO DI COSMO ET JEAN-PIERRE ARCHAMBAULT LORS DE LA RENCONTRE-DÉBAT SUR LES LOGICIELS LIBRES DU 12 JUIN 2001 À L'EPFL

résolu, un technicien peut intervenir à distance...). Au cœur de l'architecture se trouve un serveur Linux. D'autres académies expérimentent cette solution.

Des enseignants responsables des parcs informatiques de leur établissement en viennent à Linux, lassés qu'ils sont par les *plantages* à répétition ou *les fichiers élèves effacés par les copains*. La gestion des ordinateurs n'est pas indifférente à la qualité des systèmes employés. Si la télémaintenance et la mécanisation des tâches, la protection contre les agressions internes ou externes, intentionnelles ou accidentelles, la régénération des stations sont facilitées, si le système utilisé est fiable et stable, c'est autant de temps gagné pour l'évolution normale des configurations et l'aide de nature pédagogique à apporter aux enseignants de l'établissement. Et les coûts de fonctionnement en sont d'autant minorés. Il y a là des promesses à examiner avec soin, qui ne signifient pas, bien entendu, que l'on pourrait se passer de l'action de personnels compétents et formés à l'administration des parcs informatiques.

A l'autre bout de la chaîne, des écoles primaires se dotent de réseaux d'une douzaine d'ordinateurs, par exemple

L'offre logicielle libre pédagogique est abondante pour l'université mais encore insuffisante pour l'enseignement scolaire, et déséquilibrée en faveur des disciplines scientifiques. Le CNDP a entrepris un travail de recensement, examen sur un plan technico-ergonomique et documentation de l'existant. Il sera accessible sur son site Web et pourra donner lieu à des réalisations à partir du concept DémoLinux [4].

DES ENJEUX ÉDUCATIFS

Solution alternative pour les configurations des établissements scolaires, les logiciels libres représentent aussi des enjeux éducatifs et sociétaux, nonobstant le fait indiscutable et indiscuté que l'accès au code source n'est pas en lui-même d'un grand intérêt pour le simple utilisateur.

Tout le monde s'accorde à dire qu'il faut enseigner des concepts et non les *recettes* de la *nième* version d'un logiciel. Un tel objectif se réalise plus aisément dans un contexte de pluralisme technologique. Il faut habituer les élèves à la di-

versité scientifique et technique. Là où l'informatique est objet d'enseignement, l'accès au code source est incontournable quand il faut comprendre les principes et le fonctionnement des systèmes.

De nombreux travaux pédagogiques sont accessibles sur les Web académiques. Un équivalent de la licence GPL peut servir de cadre à la protection de la propriété intellectuelle de l'institution et des enseignants. La question d'une transférabilité de l'approche du libre dans le domaine des ressources éducatives est posée.

Par la nature même de leur métier et leur culture, les enseignants sont attachés à la libre diffusion à tous de la connaissance et à son partage. D'une manière générale, les logiciels libres donnent l'occasion d'une réflexion critique sur l'industrie informatique, les mécanismes de la création de richesses immatérielles, l'économie des biens informationnels, la propriété intellectuelle, et des tendances à l'œuvre dans le domaine de l'éducation.

LA SITUATION DE L'INFORMATIQUE GRAND PUBLIC

Le logiciel libre connaît une réelle ascension. Pour les ventes et mises à jour des systèmes d'exploitation serveurs, au plan mondial, Linux est devenu le concurrent direct de Windows (des parts de marché équivalentes). Il est désormais le premier système pour les serveurs Web dont le plus répandu est le logiciel libre Apache. Tous les constructeurs informatiques ont intégré le logiciel libre dans leur stratégie commerciale.

Ce développement du logiciel libre constitue pour une part une réaction à la situation de l'informatique grand public. Des bénéfices de l'ordre de 50% y sont monnaie courante, à comparer aux 3 ou 4 % de l'industrie automobile. L'on sait la course effrénée à la version suivante, avec une obligation d'achat pour rester compatibles et pouvoir échanger des fichiers, sans que cela s'accompagne le plus souvent pour le monde éducatif de la moindre justification d'ordre pédagogique. L'habitude a été prise de faire déboguer les logiciels par les clients, et de les faire payer pour des mises à jour qui ne font que corriger les défauts. L'industrie informatique est quelque peu singulière, étant la seule à ne pas garantir qu'un produit fera ce qui est annoncé (elle garantit même qu'il y a des erreurs !), et à ne pas se sentir tenue de réparer les dégâts causés par un cédérom qui aura *écrasé* les fichiers d'un utilisateur.

Ces pratiques sont directement liées à la situation de quasi monopole qui prévaut pour les systèmes d'exploitation et les progiciels. Au-delà du comportement de tel ou tel, il existe une tendance à l'instauration de monopole dans ce secteur d'activité, qui résulte de deux causes principales. D'une part, les coûts marginaux, correspondant à la production d'un exemplaire supplémentaire, se rapprochent de zéro. Contrairement à la sphère des biens matériels, où le dépassement d'un certain volume entraîne une baisse de rentabilité de l'appareil productif, le phénomène des ren-

dements croissants permet des économies d'échelle considérables pour les acteurs dominants. D'autre part, les externalités de réseau jouent un rôle prépondérant. En amont, les éditeurs ou les fabricants de composants matériels favorisent les plates-formes informatiques les plus populaires qui représentent un marché plus important, et qui le deviennent encore davantage. En aval, intervient le conformisme raisonné du client autour des produits dominants. Il y voit un gage de pérennité (on confond souvent pérennité des sociétés et de leurs produits). Il lui est effectivement plus facile de trouver du personnel formé ou de l'assistance technique. Et puis, en cas de problèmes, un directeur informatique ne se verra pas reproché d'avoir choisi Microsoft ou IBM !

L'accès au code source contrarie les politiques de *verrouillage* du marché par les acteurs dominants. Ainsi, les logiciels libres s'opposent à cette tendance au monopole et à ses inconvénients: absence de pérennité des standards et des logiciels soumis au bon vouloir des éditeurs, moindre effort de recherche et d'innovation, dépendance pour les prix et les stratégies industrielles, absence d'alternative en cas de conflit,... Ils constituent un outil de régulation de l'industrie informatique, conjointement avec les standards ouverts qui, rappelons-le, ont permis le développement d'Internet (TCP/IP, HTTP, HTML).

LA CONNAISSANCE SCIENTIFIQUE

Pour l'essentiel, un consensus existe sur la qualité des logiciels libres. D'où provient-elle? Un programme informatique est de même nature qu'une preuve mathématique. La recherche scientifique, dont le but est l'accroissement du savoir humain, est aussi une structure économique où l'on échange des biens (connaissance, renommée, attention),... Ses mécanismes moteurs et régulateurs sont fondés sur:

- la libre circulation de l'information,
- le jugement par les pairs,
- la liberté de reprendre, amender, déformer, reformuler, étendre les contributions existantes.

Ce modèle a parfaitement rempli son rôle. Or, le logiciel libre fonctionne selon ce modèle. L'équivalent de la libre circulation réside dans la publication du code source. Le débogage des programmes par des milliers de développeurs disséminés sur la planète correspond à la validation. C'est

un des points forts majeurs du libre, tant il est vrai qu'une même personne peut passer cent fois sur une erreur et ne pas la voir.

TRAVAIL COOPÉRATIF ET MODÈLE ÉCONOMIQUE

Les briques logicielles de base d'Internet sont des produits libres (SendMail, Apache,...). Inversement, la Toile rend possible un travail coopératif organisé et décentralisé, avec des in-



teractions entre développeurs et utilisateurs. Elle minimise grandement les coûts de transaction de la nécessaire coordination des différents acteurs. Elle permet de s'attaquer à des gros projets, largement au-delà des petits programmes.

Les développeurs résolvent des problèmes qui les intéressent. Ils y prennent un plaisir légitime. Internet relie des *égoïsmes* individuels pour réaliser des tâches impossibles sans une coopération soutenue. Quid de la démarche quand ce n'est plus le cas ? Se pose donc la question d'un modèle économique viable du libre. Sur quels points d'appui peut-il reposer ?

En premier lieu figure la fonction de régulation que le logiciel libre exerce dans l'intérêt général de l'industrie informatique. Viennent ensuite les intérêts particuliers. Des entreprises y trouvent leur compte en vendant du service autour d'un logiciel qu'elles fournissent gratuitement (installation, paramétrage, écriture d'applications spécifiques...). Cette démarche n'est pas si nouvelle que cela : que l'on songe au Minitel distribué gratuitement lui aussi, et engendrant moult appels téléphoniques pour des communications et des consultations en tout genre. Ou aux appareils photographiques vendus très bon marché, et aux nombreuses pellicules qu'il faut ensuite faire développer.

Les administrations, les universités et les laboratoires de recherche constituent un potentiel de création et de réalisation de premier plan. Ces entités ont tout intérêt à institutionnaliser l'approche du libre pour leurs propres besoins et activités.

D'une manière fondamentale, le phénomène du logiciel libre ne serait-il pas un révélateur de tendances à l'œuvre dans la production des biens immatériels, où le service prend le pas sur le produit lui-même et où la valeur réside non tant dans la possession d'une information pléthorique que dans la capacité à se l'approprier pour la transformer en compétence et connaissance ?

PROPRIÉTÉ INTELLECTUELLE ET ÉDUCATION

La communauté éducative s'inquiète légitimement de constater qu'au plan mondial des états et des entreprises agissent pour faire inclure l'éducation (et la santé) dans la sphère des services marchands. Ce n'est pas le cas de la France, attachée qu'elle est à son service public d'enseignement et à ses exceptions éducatives et culturelles.

Mais la marchandisation de certains secteurs éducatifs peut emprunter des voies détournées. Par exemple celle de la propriété intellectuelle du logiciel. Quelques rappels. Un programme informatique a une nature double. Information codifiée et transmissible, il est l'expression dans un langage formalisé d'un schéma de calcul constitué de l'articulation d'algorithmes. A ce titre, il peut relever du droit d'auteur traditionnel. C'est le choix qui a été fait en France et en Europe dans les années soixante-dix, pour des raisons essentiellement politiques et économiques, afin de ne pas subir la domination de l'industrie américaine et de ses nombreux brevets. Mais un programme est aussi une technologie. Compilé et associé à une machine pour la mise en œuvre d'un processus, il ne nécessite pas de la part de l'utilisateur connaissance et compréhension de son schéma. Il n'est pas seulement information sur la conduite d'une action. Il engen-

dre aussi un processus (de ce point de vue, un ordinateur avec traitement de texte ressemble à une machine à écrire). Il peut donc être considéré comme une technologie et, alors, relever du droit des brevets. En fait, il se situe quelque part entre le territoire traditionnel du brevet et celui du copyright.

En Europe, des forces déploient des efforts opiniâtres pour que le logiciel relève du droit des brevets. Ce dernier serait-il de nature à assurer un meilleur compromis entre incitation à l'innovation et diffusion de la connaissance, performance individuelle et efficacité collective ? Avec les recherches en contrefaçons qu'il autorise [5], ne devrait-on pas, au contraire, redouter une entrave à l'innovation, quand on sait que la dynamique de l'industrie du logiciel repose sur la cumulativité, l'incrémentalité et une pratique combinatoire ? Depuis quelques années, on constate une dérive. Dans un premier temps, l'Office européen des brevets de Munich a donné son accord à des brevets déposés sur des inventions comportant un programme d'ordinateur, c'est à dire incluant un élément faisant expressément partie du domaine d'exception, puis sur des procédés contenant un programme innovant. On a ensuite étendu le champ de l'industriel en y intégrant les activités commerciales, en définissant l'invention comme nouvelle solution technique d'un problème technique. Cela a permis d'accepter des brevets sur des procédés de traitement de l'information comprenant une dimension technique. Puis est venu le temps des méthodes d'organisation comportant un effet technique (par exemple l'introduction d'une simple adresse IP suffit !). A quand le tour des méthodes intellectuelles susceptibles d'être mises en œuvre de façon automatique ? Et celui des idées ? Aux Etats-Unis, on brevète des méthodes pédagogiques. Quid alors d'une méthode d'enseignement de l'écriture utilisant un traitement de texte ?

NOTES

- [1] Soit dans un programme, l'instruction conditionnelle suivante, écrite dans un langage évolué: Si $x=5$ alors $x=x+4$. Cette ligne de code source est parfaitement compréhensible: on effectue un test sur le contenu de la variable informatique x , puis, selon le résultat de cette opération de test, on procède ou non à l'affectation d'une nouvelle valeur à la variable x . Pour pouvoir être exécutée par un ordinateur, une telle instruction doit être ensuite compilée. Elle prend alors la forme d'une suite de 0 et de 1 (1110010100010...). Cette ligne de code objet, compréhensible par la machine, est par contre fortement incompréhensible pour un humain (l'opération inverse de décompilation est longue et complexe).
- [2] Par exemple, la conférence Education de la LinuxExpo 2001 a réuni 450 enseignants au CNIT à La Défense.
- [3] Offre commerciale d'éditeurs (Red Hat, Mandrake, Suse, Caldera...) composée du système d'exploitation Linux et de divers compléments.
- [4] Produit développé par Roberto Di Cosmo et ses étudiants de l'ENS Ulm pour permettre de découvrir Linux sans avoir à l'installer. Il reste en mémoire pendant son utilisation et ne modifie rien sur le poste de travail. Il peut être utilisé à la maison pour poursuivre les travaux entrepris.
- [5] Les recherches en contrefaçons sont plus que grandement facilitées par la diffusion du code source. On voit la menace que le brevet, tel qu'il existe actuellement, recèle pour le logiciel libre. ■

L'économie du Logiciel Libre

ORGANISATION COOPÉRATIVE ET INCITATION À L'INNOVATION



DOMINIQUE FORAY, DIRECTEUR DE RECHERCHE CNRS, IMRI,
UNIVERSITÉ DE PARIS-DAUPHINE ET OCDE/CERI, DOMINIQUE.FORAY@OECD.ORG
& JEAN-BENOÎT ZIMMERMANN, DIRECTEUR DE RECHERCHE CNRS,
GREQAM-EHESS, MARSEILLE, JBENOIT@EHESS.CNRS-MRS.FR *



INTRODUCTION

Les problèmes de propriété intellectuelle qui se posent à l'économie de l'innovation relèvent de trois séries de questions emboîtées. Premièrement, une connaissance nouvelle peut-elle ou doit-elle être appropriée de façon privée? Deuxièmement, si oui, sous quel régime de droit de propriété intellectuelle doit-elle être enregistrée? Et troisièmement, s'il s'agit du régime des brevets, quelle pratique de création et d'usage du brevet s'avérera-t-elle favorable à l'établissement d'un bon équilibre entre la protection de l'inventeur et l'usage social de son invention? Il est remarquable de constater que le logiciel, comme d'ailleurs les créations génétiques, sont des objets pour lesquels il n'y a de réponse assurée à aucune de ces trois questions. La première question rend compte de l'opposition entre logiciel propriétaire et logiciel libre et les termes de cette question sont principalement économiques comme nous le verrons par la suite: ce n'est pas le *peut-il être* qui pose problème comme dans le cas des créations génétiques (question notamment éthique) mais c'est plutôt le *doit-il être* qui fait question. La deuxième question rend compte de l'incertitude sur le régime de propriété intellectuelle, qu'il convient d'activer dès lors que l'on a répondu oui à la question sur l'appropriation privée. La protection par brevet est-elle préférable à la protection par droit d'auteur ou bien, compte tenu du caractère insatisfaisant de ces deux régimes pour les logiciels, ne vaudrait-il pas mieux créer un droit *sui generis*, ainsi qu'on l'a fait notamment pour les variétés végétales? Enfin, la troisième question fait référence aux situations de blocage qui peuvent découler d'une mauvaise utilisation du brevet: soit parce que la surface du brevet est très étendue et bloque les projets d'innovation ultérieurs sur le domaine, soit parce qu'un nombre très important de brevets couvre les composants élémentaires d'une innovation, ce qui disperse la base de connaissances et ralentit le développement d'un produit final par un agent particulier.

Cet article porte principalement sur les deux premières questions et les traitera dans un ordre inversé. C'est parce qu'aucun des régimes de droit de propriété intellectuelle n'offre une solution totalement satisfaisante – et cela compte tenu de ce que l'usage de la propriété intellectuelle nous situe de toute façon dans un cadre de solutions de second rang (section 1) – que l'on *remontera* vers la question du logiciel libre. Après en avoir identifié les bonnes propriétés du point de vue de l'économie de l'innovation (section 2), nous évoquerons les fondements institutionnels du logiciel libre (section 3) puis nous examinerons le rôle des incitations individuelles dans ce modèle (sections 4 et 5).

1. LES PROBLÈMES DE LA PROPRIÉTÉ INTELLECTUELLE DANS LE DOMAINE DU LOGICIEL

La question de la protection de la propriété intellectuelle pour le logiciel s'est posée à partir des années 70, avec son accès au statut de marchandise, dès lors que la législation antitrust américaine avait mis un coup d'arrêt aux pratiques de facturation liée introduites par IBM. Longtemps classé au rang des activités de services ou considéré comme produit joint de la construction d'ordinateurs, le logiciel est donc considéré aujourd'hui comme une activité maîtresse et autonome au sein de l'industrie informatique. Il reste cependant une activité industrielle très particulière, essentiellement vouée au développement de technologies de *process* de calcul ou de commande et à leur mise en forme opérationnelle, en conformité avec des standards opératoires: formatage des données, interfaces d'entrée-sortie, protocoles de communication, interfaces homme-machine. Marchandise un peu particulière pour l'analyse économique, il peut s'apparenter à de la connaissance, bien qu'en même temps sa mise en œuvre par un utilisateur donné ne requière pas de la part de celui-ci une appropriation effective. Mais,

* Article à paraître en 2002 dans le numéro spécial de la Revue Economique (Paris) consacré à *la nouvelle économie*.

compilé, c'est-à-dire traduit en langage-machine, sous la forme d'un *code-objet*, il revêt un caractère particulier d'autonomie; il n'est plus seulement information sur la conduite d'un processus, d'une action, il est lui-même action, il est processus de coordination de stations logiques (les modules d'une architecture informatique), en vue de réaliser une tâche de calcul, de traitement et de manipulation de données, de commande, c'est-à-dire une tâche de nature informationnelle (Zimmermann, 1995).

La protection de la propriété intellectuelle répond à une double préoccupation. Dans un sens, elle constitue une incitation à l'innovation, en octroyant à l'inventeur un pouvoir de monopole temporaire pour l'exploitation industrielle et commerciale de son invention. Dans l'autre, elle vise, du moins en ce qui concerne le système des brevets, une meilleure diffusion de la connaissance, de par l'obligation qui est faite à l'inventeur, pour obtenir la protection, de révéler les principes et la nature du produit ou du procédé sur lequel porte la requête. Ces deux ordres opposés de préoccupations constituent le fondement de ce que l'on désigne parfois comme le *dilemme schumpétérien*, auquel une solution est généralement recherchée en termes d'étendue¹ et de durée optimale de la protection en vue de *constituer un facteur d'incitation suffisant ex-ante sans trop entraver l'imitation et la diffusion ex-post* (Heraud, 1995). Une telle solution ne peut être considérée comme satisfaisante que dans un contexte défini en fonction des caractéristiques de l'industrie et de la dynamique de l'innovation, tout particulièrement en ce qui concerne le rythme de l'innovation et ses propriétés cumulatives.

S'agissant du logiciel, le problème principal réside dans l'incertitude sur la classe de droit de propriété intellectuelle (droit d'auteur ou brevet) qui serait la plus adaptée aux caractéristiques de l'objet. Selon Lucas (1987), *«le programme qui est au cœur du logiciel se caractérise d'abord par son contenu en tant que procédé permettant de tirer parti des ressources de la machine en vue d'un résultat déterminé. A ce titre, sa protection pose un problème de brevetabilité. Mais, en même temps, il se présente en lui-même, apparemment au moins, comme une œuvre de l'esprit susceptible de donner prise au droit d'auteur (...). De là vient qu'il puisse a priori prétendre à la fois au bénéfice du brevet d'invention et à celui du droit d'auteur. De là vient aussi qu'il ne puisse trouver commodément sa place ni dans l'une ni dans l'autre de ces deux branches de la propriété industrielle, ce qui fait toute la différence en la matière.»*

Néanmoins, plutôt que de concevoir un cadre juridique spécifique, l'orientation générale, aux Etats-Unis,



puis en Europe et au Japon, a consacré la prédominance du droit d'auteur. Cette unanimité s'accommode toutefois de variantes, relatives tant aux contextes juridiques nationaux de la propriété intellectuelle, qu'au rôle et à la place de la jurisprudence (le droit anglo-saxon, beaucoup moins codifié que le droit latin, lui confère à cet égard une fonction de premier ordre), voire aux attitudes culturelles vis-à-vis de la propriété intellectuelle.

En Europe, et notamment en France, le système juridique de la propriété intellectuelle a jusqu'ici exclu assez clairement le logiciel du champ d'applicabilité des brevets, hormis dans le cadre d'un procédé industriel brevetable, la protection restant alors limitée au cadre du procédé de référence. Aux Etats-Unis, en revanche, la jurisprudence a ouvert la voie, dans le courant des années quatre-vingt-dix, à une dérive vers un recours accru au système des brevets, suscitant une opposition radicale entre partisans et opposants de la brevetabilité du logiciel. Depuis lors, le débat s'est ouvert en Europe, en vue de la nécessaire détermination d'une approche commune. Une étude a été réalisée à la demande de la Commission Européenne sur l'impact économique de la brevetabilité

des programmes informatiques (Hart, Holmes et Reid, 2000). Une consultation ouverte a été lancée, via l'Internet, parallèlement à la Conférence Intergouvernementale de Munich de novembre 2000.

En soi, l'observation des effets de ces pratiques sur le comportement des acteurs de l'industrie du logiciel aux Etats-Unis semble indiquer que la brevetabilité du logiciel conduirait à une forme de cloisonnement des savoirs et des procédures, qui interdirait ou du moins freinerait toute pratique combinatoire. Parmi la multitude de brevets accordés, on trouve

un grand nombre de procédures, voire d'algorithmes. Beaucoup d'entre eux apparaissent très éloignés des critères de nouveauté et d'originalité qui conditionnent, en principe, la délivrance d'un brevet. Ils s'apparentent souvent à des procédures conventionnelles et largement répandues, à l'utilisation desquelles les détenteurs de brevets pourraient donc désormais s'opposer. Un des aspects principaux du problème est que le législateur ne peut statuer ici de manière acceptable en l'absence d'un état de l'art patenté, alors même que la législation américaine est supposée appliquer le principe du *first to invent*, par opposition à la plupart des autres régions du monde qui pratiquent celui du *first to file* (Heraud, 1995).

Dans des travaux récents, Bessen et Maskin (2000) mettent en avant l'argument que, dans une industrie comme le logiciel, où l'innovation est à la fois séquentielle² et complémentaire³, l'introduction d'un système

1 Selon Scotchmer (1991), le système nord-américain ayant tendance à octroyer une forte étendue de la protection au premier inventeur, risque de réduire les incitations pour les innovations suivantes, secondaires ou induites.

2 Dans le sens du caractère cumulatif du progrès technique, chaque invention successive prenant appui sur les précédentes.

3 Dans le sens où des recherches menées en parallèle et avec des approches différentes sur un même sujet accroissent la probabilité de réussite à horizon temporel donné.

fort de propriété intellectuelle fondé sur les brevets aurait un effet inhibiteur sur la dynamique de l'innovation. En revanche, un régime de faible protection, parce qu'il ouvre des possibilités d'imitation, tend paradoxalement à encourager les efforts de recherche et d'innovation. Ces résultats ne tiennent certes pas compte de la procédure particulière des brevets qui permet à l'auteur d'une innovation de perfectionnement d'obtenir une licence d'exploitation d'un brevet pionnier antérieur. L'apport du modèle théorique semble cependant bien corroboré par l'observation de l'industrie américaine du logiciel, qui a connu une stagnation, voire une régression des efforts de R&D, au moment même où la jurisprudence accordait de plus en plus largement la délivrance de brevets pour des programmes d'ordinateurs. Et cette tendance régressive apparaissait particulièrement marquée pour les firmes qui brevetaient le plus.

Mais, de son côté, la législation du droit d'auteur, appliquée au logiciel, a cette particularité qu'elle procure la protection de l'œuvre sans obliger à en dévoiler le contenu. La double nature de l'expression d'un programme, sous la forme d'un code-source explicite et d'un code-objet, après compilation, en langage machine, permet de disposer des potentialités du produit logiciel sans nécessiter l'accès à son expression explicite. Par conséquent, l'immense majorité des éditeurs commercialisent les programmes sous la seule version de leur code-objet et gardent secret le code-source. *«La création étant fonctionnelle, l'utilisateur peut obtenir le résultat attendu: les fonctions ou, à l'américaine, les fonctionnalités, sans avoir à se préoccuper de ce que les juristes ont élu comme objet de droit: la forme»* (M.Vivant, 1993). La question immédiatement corollaire est celle de l'ingénierie inverse, ou décompilation, qui vise à remonter au code-source à partir du code-objet. Malgré une certaine diversité des législations et des jurisprudences, la limite de tolérance en la matière, telle que l'a exprimée la Directive Européenne, est celle de la révélation des interfaces en vue de permettre l'interopérabilité des programmes, à condition toutefois que l'éditeur du logiciel n'ait pas accepté de révéler par lui-même ces spécifications d'interfaces. Au delà de cette limite, la violation du droit d'auteur est considérée comme avérée. Ainsi le droit d'auteur, appliqué aux logiciels ne répond-il qu'à l'une des deux préoccupations à l'origine de la protection de la propriété industrielle, la protection de l'inventeur et non la diffusion des idées. *«Pour la première fois depuis Sybaris, cinq siècles avant Jésus-Christ, qui dans sa loi sur l'appropriation des recettes de cuisine en imposait la divulgation au public, voici donc réconciliés propriété et secret!»* (M.Vivant, 1993).

En réalité, la véritable opposition n'est pas celle entre brevets et droit d'auteur, mais entre deux modèles de développement du logiciel, deux approches fondées sur des principes radicalement opposés et donc prédisposées à diverger. Ces deux modèles sont ceux fondés sur l'approche marchande et de l'appropriabilité, d'une part, et ceux fondés sur la culture de la *connaissance pure* et de la

coopération, d'autre part. Cette opposition est précisément celle qui divisait Bill Gates et ses collègues académiques dans les années soixante-dix au MIT. Le premier ne concevait pas que d'autres puissent utiliser librement le fruit de son travail de conception. Ses collègues, baignés de culture scientifique et communautaire, considéraient en revanche qu'un code est de la connaissance pure et constitue à ce titre un bien public qui ne peut être approprié de manière privative (Smets et Faucon, 1999). Si Bill Gates a connu la carrière industrielle que l'on sait, les seconds, qui s'inscrivent dans la tradition et la logique de la science ouverte (Dasgupta et David, 1994), ont été à l'origine du logiciel libre. Mais ils se situent aussi dans la tradition de tout un milieu de développeurs qui, même intégré dans un contexte d'entreprises, a fondé ses pratiques sur la diffusion des idées et le caractère cumulatif du progrès technique, aussi bien en matière de concepts et d'outils, que d'algorithmes de résolution de problèmes ou de méthodes et de styles de programmation.

2. LES BONNES PROPRIÉTÉS DU LOGICIEL LIBRE

Un logiciel libre est un logiciel dont le code-source, c'est-à-dire la série d'instructions qui forme le programme avant la compilation, est rendu ouvertement disponible et ne peut faire l'objet d'une appropriation privative. Le développement des logiciels libres est fondé sur le volontariat et le bénévolat des participants, dans un mode d'organisation coopératif qui s'appuie largement sur les commodités organisationnelles issues d'Internet⁴. Le produit phare en est aujourd'hui le système d'exploitation Linux, inspiré d'Unix et portable sur des architectures de micro-ordinateurs. Mais beaucoup d'autres produits sont également disponibles auprès d'entreprises spécialisées ou même, le plus souvent, téléchargeables gratuitement sur Internet: serveurs de site Web, utilitaires bureautiques, scientifiques, de traitement d'image,...

Ce que nous souhaitons montrer dans cette section est que ce qui caractérise le modèle du logiciel libre n'est pas principalement la gratuité, qui ne constitue d'ailleurs pas une règle, mais surtout la liberté de modifier et d'améliorer les versions existantes, ainsi que l'impossibilité de s'approprier les améliorations apportées. L'économie du logiciel libre est donc d'abord un modèle d'innovation, d'accumulation de la connaissance et de recombinaison des savoirs et non pas une stratégie de marketing de type *offre gratuite de produits*.

Le fait de donner à l'utilisateur l'accès au code-source permet d'engendrer des effets très importants d'apprentissage par l'usage, c'est à dire d'exploiter au mieux une très grande intelligence distribuée: les millions d'usagers qui révèlent des problèmes et les milliers de programmeurs qui trouvent comment les éliminer. D'après les termes de la Free Software Foundation, chacun peut uti-

4 Voir à ce propos D. Desbois, N. Jullien, T. Pénard, A. Poulain-Maubant, J. Vétois et J.-B. Zimmermann (Eds.) (1999).

liser le code et le modifier, à la condition de communiquer la modification à l'organisation, pour qu'elle soit vérifiée et évaluée. On retrouve ici *les bonnes propriétés* de la distribution de la connaissance et des systèmes de savoir ouvert: seule la circulation rapide et élargie des savoirs permet de bénéficier du potentiel unique d'un très grand nombre d'individus compétents. D'un point de vue concret, une distribution rapide de la connaissance facilite la coordination entre les agents, elle diminue les risques de duplication entre projets de recherche et elle permet de concentrer l'apprentissage sur les *meilleures* inventions. Elle constitue en outre une assurance de qualité, puisqu'une fois la connaissance produite, elle est testée et donc vérifiée par de nombreux agents. Enfin, en propageant la connaissance au sein d'une population diversifiée de chercheurs et d'entrepreneurs, elle accroît la probabilité de découvertes et d'inventions ultérieures, en même temps qu'elle diminue le risque que cette connaissance soit détenue par des agents incapables d'en exploiter les potentialités (David et Foray, 1995).

Ces bonnes propriétés du savoir ouvert sont amplifiées dans le cas des logiciels libres:

- premièrement, un logiciel constitue un objet scientifique ou technologique complexe, qui est donc caractérisé par des processus d'apprentissage quasi-illimités: un système composé de milliers de développeurs travaillant sur le même logiciel durant une longue période restera longtemps, presque indéfiniment, dans une phase de rendements croissants – ce qui ne serait pas le cas d'un système de milliers d'ingénieurs travaillant à l'amélioration d'une brouette;
- deuxièmement, un logiciel est exprimé sous la forme d'un ensemble d'instructions codifiées qui circulent parfaitement sur le réseau électronique. Ainsi, la circulation des améliorations apportées est rapide, parfaite, et son coût marginal est quasi-nul. Ceci accroît l'efficacité globale du processus de recherche collective. Ceci accroît aussi l'incitation des agents à envoyer de l'information. Comme l'ont bien montré Lakhani et Von Hippel (2000) dans leur étude empirique du système Apache, si tant d'agents acceptent de coopérer, c'est notamment parce que le temps passé à envoyer de l'information ne dépasse pas 5 minutes.
- troisièmement, les logiciels appartiennent à une certaine classe de technologie ayant la propriété particulière de réduire, voir même d'annuler la distance entre producteurs et consommateurs (Quah, 1999). Les millions d'utilisateurs qui révèlent les problèmes sont, pour une part, les développeurs qui proposeront les solutions. Le va-et-vient entre identification des pro-

blèmes et formulation des solutions est donc quasi-instantané. Là encore, Lakhani et Von Hippel (2000) ont un résultat intéressant: les gens s'entraident parce que la solution recherchée par l'un est *sur étagère*. Autrement dit, elle est connue par un autre qui n'aura donc pas à consentir un effort très grand pour retrouver et envoyer celle-ci.

Ces différents aspects ont une traduction évidente et indiscutable en termes de performance des produits. Le taux d'innovation, la qualité et la fiabilité sont nettement supérieurs à ce que l'on trouve dans le monde des logiciels propriétaires. Si les évidences empiriques sur ce point restent parcellaires, le simple fait que de nombreuses entreprises et administrations se rallient au monde du logiciel libre est un bon indicateur du potentiel des performances attendues.

Pour ces raisons, le système Linux ne doit pas simplement être analysé en tant que mode privilégié d'expression des convictions éthiques et altruistes des individus, ou même d'un sentiment communautaire, mais il doit être surtout vu comme un mécanisme générateur d'efficacité économique.

3. LES FONDEMENTS INSTITUTIONNELS D'UN MODÈLE COOPÉRATIF

Le mode de développement coopératif du logiciel libre, fréquemment désigné comme le modèle du *bazar* (Raymond, 1998), prend appui sur le potentiel de diffusion et de communication offert par l'Internet et la mutualisation de ressources qu'il autorise. Il est fondé sur le principe de la disponibilité des codes-source, afin de permettre à n'importe quel développeur de réaliser toutes modifications ou améliorations qui pourraient lui sembler utiles. Mais ces modifications n'ont d'intérêt, à un niveau collectif, que si leur auteur les rend à son tour publiques, afin qu'elles puissent être éventuellement intégrées à la construction d'ensemble⁵. Le fonctionnement effectif d'un mode d'innovation continu, fondé sur une logique de don/contre-don (Dang Nguyen et Pénard, 2001), nécessitait de se doter des outils juridiques minimaux indispensables pour protéger la communauté des développeurs contre les comportements opportunistes, et notamment contre l'appropriation privée de tout ou partie du code⁶. Pour éviter cet écueil, qui est celui du logiciel du domaine public, il était nécessaire de rentrer dans le champ du droit d'auteur, avec cependant un positionnement très différent, voire opposé à celui du copyright fondé sur la propriété privée de l'expression. De ces préoccupations sont nés les principes du *copyLeft* institués par la Free Software Foundation. Ces principes sont

5 Pour ce qui concerne Linux, le code source initial représentait environ 50 000 lignes de programme; il dépasse aujourd'hui le million. Un petit groupe informel d'experts sélectionne et valide les propositions d'amendements ou d'ajouts. Mais ce type de coordination informelle, sur un modèle associatif, risque d'atteindre rapidement ses limites, avec la croissance du nombre des participants, d'autant qu'il n'y a ni financement ni rémunération des travaux de programmation et de coordination. (Ganagé et Bonnet, 1998).

6 Se prémunir contre l'appropriation privée du code n'exclut pas pour autant une éventuelle exploitation marchande de ce code.

ceux de la libre utilisation du code et de sa libre modification, sous réserve d'en communiquer la teneur à l'organisation en vue de sa *vérification* et sa *labélisation*. Ainsi, dans le cas de Linux, «*la décision ultime d'intégrer ou non les changements est du ressort de l'un des dictateurs bienveillants qui dirigent le secteur concerné par le projet Linux*». (Alper, 1999).

En réalité, le besoin de principes de cet ordre s'est imposé dès lors que se mettait en place un premier projet collectif de développement, GNU⁷, en 1983. La nécessité de pourvoir à un cadre juridique susceptible de garantir le respect des principes du développement coopératif a donné naissance à la licence GPL ou GNU-*General Public License*⁸. Le principe fondamental en est que, les codes concernés étant libres, chaque programme qui intègre des lignes de code GPL doit aussi être disponible sous licence GPL. Ainsi, les auteurs n'abandonnent pas leurs droits, mais la seule rente de monopole que ces droits autoriseraient dans un régime de copyright. Le logiciel reste de la sorte la propriété de ses créateurs. Ceux-ci autorisent quiconque à en faire usage (modifications, améliorations, compléments...), sous la seule condition que toute nouvelle version puisse, elle aussi, circuler librement.

Ceci n'est pas exclusif d'une éventuelle commercialisation de ces programmes et ne limite donc pas le cadre du logiciel libre à la sphère non-marchande. Mieux encore, certaines entreprises ont fondé leurs stratégies de développement sur les principes de développement coopératif et de libre accès aux codes-source, élargissant ainsi le cadre conceptuel du logiciel libre à un champ que l'on désigne fréquemment comme celui des OSS (*Open Source Software*). Cet élargissement peut, soit porter sur des activités de développement, d'édition et de distribution de logiciels libres, à l'instar de la société américaine Redhat Software; soit conduire à une ouverture des sources de certains produits de la part de sociétés de logiciel traditionnelles. Ainsi, un logiciel peut fort bien devenir *libre* après avoir été protégé et payant, comme l'a fait Netscape en 1998 pour son produit *Communicator*, dont le code-source est librement diffusé. Cette stratégie avait pour objectif de chercher à contrer Microsoft sur son propre terrain, en mobilisant un important effectif de programmeurs sur un produit-clef, afin de l'imposer comme standard de fait. La logique économique pour Netscape était de réaliser ensuite ses gains sur les produits complémentaires, lesquels peuvent sans difficulté rester de l'ordre des architectures propriétaires.

Cette implication des activités marchandes dans le domaine du libre a dicté la nécessité de concevoir un élargissement du cadre juridique du *copyLeft*, au delà de la trop stricte licence GNU-GPL. De là est né un ensemble de licences plus hybrides, visant à concilier développement coopératif et intérêts privés au sein des OSS. Les modèles abondent et traduisent une variété de situations et de stratégies particulières, qui ne peuvent s'intégrer dans aucun régime universel. Ils correspondent à des positionnements divers qui associent, en des proportions et des modes variés, la logique du copyright à celle du *copyleft* (Smets et Faucon, 1999).

Ainsi, dans son opération d'ouverture de *Communicator*, Netscape a mis au point deux licences complémentaires: NPL (*Netscape Public License*), qui permet d'intégrer les nouveaux développements à ses serveurs sans nécessiter de faire entrer ces derniers dans le champ du *copyleft*, et MPL (*Mozilla Public License*)⁹, qui couvre le développement de nouveaux modules. De la sorte, toute modification à partir du code source de *Communicator* doit être publiée, mais Netscape se réserve le droit d'intégrer dans *Communicator* des modules propriétaires et se garde de contraindre

son utilisation dans sa propre gamme de produits. De la même façon, Sun a créé la licence SCSL (*Sun Community Source License*) pour *libérer* son protocole de communication *Jini*. De cette manière, il se réserve l'exclusivité de la certification en compatibilité de tout produit issu de *Jini* et destiné à une exploitation commerciale. D'autres, comme Novell ou IBM, ont suivi des voies similaires et de multiples variantes combinent les droits d'utilisation ou d'intégration des OSS avec les logiciels propriétaires et leur usage marchand et non-marchand. L'idée générale est au fond de rendre compatible un modèle d'accès libre à la technologie, favorisant la progression du savoir et l'adoption du standard, avec un modèle d'exclusivité sur certains éléments, qui garantit un certain niveau de revenus.

4. L'INCITATION DANS UN SYSTÈME D'INNOVATION COLLECTIVE

Le modèle du logiciel libre aborde aujourd'hui le virage de son entrée dans le monde marchand et de son élargissement au marché grand public. Ce virage est sans doute particulièrement délicat, car il découle du succès, voire de l'engouement, que les logiciels libre suscitent



⁷ **GNU is Not Unix**, projet développé en vue de créer une plate-forme libre de type Unix. Ce projet s'inscrivait, de manière prémonitoire, en opposition aux réflexes propriétaires qui allaient, dès la fin des années quatre-vingt, reprendre le dessus dans le monde Unix et en faire éclater le potentiel d'ouverture.

⁸ Pour une analyse fine des fondements juridiques de la GPL, voir Clément-Fontaine (1999).

⁹ Mozilla est le nom que Netscape a donné au projet de développement coopératif né de l'ouverture de *Communicator*.

depuis peu auprès d'une catégorie élargie d'utilisateurs. Jusqu'à une période récente, le développement du libre était souvent considéré comme une affaire d'informaticiens, dans la mesure où il ne débordait guère de cette communauté virtuelle d'utilisateurs-développeurs sur laquelle se fonde le modèle du *bazar*. Or voilà que le succès du libre finit par atteindre la catégorie des simples utilisateurs, en commençant par la part la plus dynamique d'entre eux, souvent très jeune, férue d'Internet et qui contribue à diffuser, à la fois chez les particuliers et dans les entreprises, des concepts innovants, des pratiques en évolution constante, des produits et des standards fondés sur des règles nouvelles. A travers cette catégorie d'utilisateurs éclairés, c'est le marché grand public (particuliers et entreprises), un marché de masse, qui commence à s'ouvrir au monde du libre. L'émergence de cette nouvelle catégorie d'utilisateurs, ainsi que l'intérêt du monde marchand pour le modèle du libre, traduisent une situation nouvelle où des agents sont en mesure de réaliser des bénéfices privés par la commercialisation ou l'utilisation des résultats des efforts collectifs de la communauté des développeurs dont la participation n'est, à la base, fondée sur aucune rémunération. Ceci aggrave bien entendu, pour ces développeurs, les problèmes d'incitation.

L'économie de la science et de la technologie a analysé de nombreux systèmes de *savoirs libres*, faisant référence, soit à des situations locales et historiques –les Canuts lyonnais (Foray et Hilaire Perez, 2000), la métallurgie du Lancashire (Allen, 1983)–, soit à des situations générales –la science ouverte (Dasgupta et David, 1994). Or, dans tous ces cas, on s'aperçoit qu'au delà des bonnes propriétés de la circulation rapide et élargie des savoirs, qui confèrent au système son efficacité économique, s'impose toujours une dimension d'incitation individuelle. Celle-ci requiert le fonctionnement de mécanismes visant à offrir un crédit aux inventeurs, sans que ce crédit ne se traduise en droit d'exclusivité. En d'autres termes, un système de savoir ouvert requiert des mécanismes d'incitation, compatibles avec le principe de partage et de circulation des inventions.

C'est le système de la règle de priorité dans le domaine de la science ouverte, qui donne à celui qui invente et qui publie son invention un droit de propriété moral. Ce droit ne se concrétise pas par une situation

d'exclusivité, mais il est à la base de la constitution d'un capital de réputation, élément décisif pour obtenir des subventions ou gagner un prix. La compatibilité entre le caractère de bien public de la connaissance et le mécanisme d'incitation est ainsi assurée par ce mécanisme remarquable, qui permet de créer un actif privé, une forme de propriété intellectuelle, qui résulte de l'acte même de renoncer à la possession exclusive de la nouvelle connaissance. En ce sens, la règle de priorité offre des incitations non marchandes (elle crée des contextes de course ou de tournoi) à la production de biens publics.

Dans le cas de la soierie lyonnaise du XVIII^{ème} siècle, on accordait une prime au Canut qui inventait un nouveau procédé, en échange de la libre circulation de l'invention. En outre, le montant de la prime était calculé en fonction de la participation de l'inventeur lui-même à la diffusion et à l'adoption par les autres de son invention. Là encore, il y a compatibilité entre incitations individuelles et mécanisme collectif. Cependant, ce système n'évite pas la question difficile de l'estimation ex ante de la valeur sociale de l'invention.

Dans le monde du logiciel libre, l'effort collectif d'amélioration d'une base de logiciels engendre pour chacun une

externalité d'usage. Cette externalité est générale au sein de la communauté des utilisateurs-développeurs, en ce sens qu'elle équivaut à la production d'un bien public non-rival et non-exclusif. Si elle constitue, à une échelle collective, une incitation à coopérer, elle engendre aussi à l'échelle individuelle, et lorsqu'elle est considérée isolément, une incitation à la défection et au comportement opportuniste (le bénéfice de l'externalité n'est pas contingent de la coopération). Indispensables à sa viabilité, les incitations individuelles sont donc présentes et au cœur de la dynamique du modèle. Au delà de l'éthique académique (sentiment de contribuer à une œuvre collective), elles sont principalement de deux ordres: l'apprentissage et la réputation. Une condition supplémentaire tient au faible coût marginal du travail exigé pour contribuer à l'entreprise coopérative.

Le premier ordre d'incitations est relatif aux effets d'apprentissage. La programmation est un art non stabilisé et dont les méthodes restent encore fondées sur des procédures et méthodes relativement artisanales et peu codifiées, y compris au sein des organisations productives des grands éditeurs¹⁰. Le fait, pour un développeur,



10 Zimmermann (1998)

de suivre et de participer aux efforts de la communauté du libre, a pour effet en retour de contribuer à améliorer son propre niveau de compétence informatique, et ceci pour au moins deux raisons. La première relève d'un apprentissage par la pratique, issu d'une participation à un effort collectif de développement. La seconde résulte de la confrontation avec un code qui, lui-même, résulte d'une multiplicité de contributions relevant d'une diversité de savoirs, de méthodes et de styles de programmation. Cet apprentissage par interaction constitue une source d'amélioration individuelle de la compétence du programmeur, bien au-delà de ce que celui-ci pourrait attendre de sa participation à une équipe plus traditionnelle de développement, fût-ce même au sein de grandes entreprises, dont l'organisation reste à ce jour cloisonnée et fractionnée en petites équipes. L'incitation à coopérer est claire pour le développeur individuel, dans la mesure où il s'agit pour lui de rester à la hauteur de l'évolution des savoirs, et ce d'autant plus qu'il se situe dans le contexte d'un jeu répété.

Le deuxième ordre d'incitations joue un rôle pivot dans notre propos. Il est relatif aux effets individuels de réputation¹¹ et fonctionne de manière très similaire à celui en vigueur dans le milieu académique. Il résulte d'un processus de reconnaissance par les pairs, dans la mesure où les modifications et améliorations acceptées et labélisées dans un logiciel libre comportent la signature explicite de leur auteur, qui apparaît comme telle dans les copies en circulation du programme. Ici, cependant, ce capital de reconnaissance a la vertu essentielle de pouvoir être converti sur un mode pécuniaire, à travers une embauche dans une entreprise où un accès privilégié à des sources de financement, à la différence du monde académique où cet objectif à terme reste une modalité secondaire.

On pourrait rassembler enfin dans un ensemble de conditions favorables, les facteurs qui contribuent à la minimisation du coût marginal de l'effort nécessaire pour contribuer à l'entreprise coopérative. Lakhani et von Hippel (2000) montrent l'importance de ces conditions, dans le cas d'un système en ligne d'assistance entre usagers. Le système fonctionne, non seulement parce qu'il y a des bénéfices individuels (apprentissage et réputation) mais aussi parce qu'il n'y a pratiquement pas de coût: il y a dans le système au moins un usager pour lequel l'effort de trouver la solution demandée par un autre usager est minime (la probabilité que la solution soit disponible quelque part dans le système est très grande) et le coût de transmission de cette solution est quasi-nul. L'effet Internet (au sens d'une minimisation du coût marginal de stockage et de transmission de l'information) est ici très important.

Dans le contexte qui prévalait jusqu'à une période récente, les individus qui tiraient bénéfice de l'utilisation des logiciels libres étaient, pour l'essentiel, ceux-là mêmes qui participaient à leur développement. La participation à l'effort d'amélioration de la base de logiciels supposait, pour chacun, que les bénéfices tirés de l'usage de cette base et les gains espérés en termes d'apprentissage et de réputation se cumulent en un avantage suffisamment élevé, pour compenser le coût minime de la participation. La dynamique interne du bazar montre que cette condition pouvait être considérée comme généralement satisfaite sans que la présence de comportements opportunistes ne mette en péril la viabilité du système¹². Nous montrerons à la section 5, sur la base d'un modèle de coalition très simplifié, que l'équilibre d'une telle configuration correspond à des conditions internes et externes assez facilement réalisables. Nous nous interrogerons également sur les effets de taille et montrerons qu'une simple hypothèse de stricte concavité dans la production du bien public engendre un effet limite sur la taille qui justifie la nécessité d'organiser la communauté du libre autour de projets dans lesquels ne tendent à s'investir qu'une fraction du total des développeurs.

Mais la principale question qui nous préoccupe ici est celle de savoir ce qu'il advient de cet équilibre, dans un contexte où un nombre croissant d'utilisateurs simples bénéficient des efforts de développement de la communauté du libre, sans être redevables d'aucune contrepartie, notamment sur le plan pécuniaire. Cet élargissement de la population concernée résulte d'un ensemble de trois facteurs complémentaires. Il s'agit, en premier lieu, de l'arrivée à un certain niveau de maturité des produits du logiciel libre; les produits sont dorénavant très performants et d'un niveau de fiabilité très élevé, du fait de la capacité de test inégalée que représente le modèle du bazar comparativement à toute structure d'entreprise. En deuxième lieu, cet élargissement est porté par l'extraordinaire moyen de diffusion que représente Internet et qui surajoute cette capacité aux propriétés d'interconnexion de la communauté des développeurs. En dernier lieu, la constitution d'entreprises commerciales, vouées à la distribution de logiciels libres édités avec des *designs* plus élaborés, des interfaces-utilisateur, des manuels d'utilisation et des utilitaires d'aide, constitue un facteur décisif pour une catégorie d'utilisateurs peu technicienne et peu rompue à la navigation sur Internet. Ce troisième type d'acteurs dans le monde du logiciel libre trouve à concilier logique commerciale et principes de non-appropriation, en fondant sa rentabilité sur la vente de services liés: *hot line*, mises à jour, *débogage*, ingénierie de systèmes d'information, formation, ... Il y aurait lieu, en outre, d'ajouter à ces trois facteurs explicatifs un qua-

11 Lerner et Tirole (2000) regroupent, au sein d'une même catégorie dite *signal incentives*, deux types d'effets distincts quoique difficiles à distinguer, d'une part les effets de réputation et d'autre part les effets de satisfaction (ego gratification) directement liés à la reconnaissance par les pairs.

12 Dans un article récent sur les comportements coopératifs dans la science ouverte, P. David (1997) développe un modèle d'interactions stochastiques entre individus rationnels, engagés dans un processus continu d'observation expérimentale, échange d'informations et révision de choix. Il montre que le fonctionnement coopératif du réseau peut tolérer les comportements opportunistes. Autrement dit, les performances cognitives du réseau (la capacité à produire collectivement un énoncé scientifique, tout en préservant la diversité des opinions) sont assurées tant que le nombre de comportements opportunistes est maintenu en dessous d'un seuil critique.

trième argument, qui tient à la logique de l'affrontement entre deux modèles de production radicalement opposés. Dans le combat que mènent bon nombre de *militants* du libre contre l'appropriation privée des codes et contre la structure oligopolistique de l'industrie du logiciel, l'adoption des logiciels libres par un nombre sans cesse croissant d'utilisateurs constitue un indicateur pertinent, en termes de part de marché, du terrain gagné.

On comprend facilement que la possibilité, pour un nombre croissant d'agents, de tirer des revenus conséquents de la commercialisation et de l'utilisation de logiciels libres, sans avoir contribué à leur développement, remet en cause de manière radicale la tolérance du système aux comportements opportunistes. Tout d'abord l'élargissement d'audience, auquel fait face aujourd'hui le libre, constitue un clair facteur de reconnaissance, donc de succès, et il serait limitatif de n'y voir que la multiplication de comportements opportunistes. Au niveau de chaque programmeur, ensuite, les succès et la reconnaissance du libre confortent sans aucun doute les effets de réputation dont bénéficient les développeurs, mais ils engendrent aussi une insatisfaction relativement à l'absence de rémunération des efforts consentis. Cette frustration, dont l'intensité s'accroît avec l'élargissement du *marché* du libre, finit par peser, dans l'arbitrage des individus, d'autant plus fort que leurs attentes en matière d'apprentissage et de réputation sont peu élevées. Nous montrerons dans la section 5 que les individus le plus facilement en manque d'incitations sont précisément ceux qui offrent le plus fort niveau de compétences. Leur défection risquerait de précipiter le système du bazar dans une dynamique de démissions et d'effondrement hors de tout équilibre. C'est le succès du logiciel libre qui en aurait alors engendré la chute.

Maintenir l'équilibre, alors même que les adopteurs et bénéficiaires du logiciel libre sont de plus en plus nombreux, suppose par conséquent l'instauration de mécanismes nouveaux susceptibles de compenser les effets de frustration à l'origine des défections. Cette compensation d'incitation peut, de la manière la plus directe, prendre la forme d'une incitation pécuniaire, par le biais d'une rémunération. Deux modes alternatifs et complémentaires s'imposent. Le premier, qui est déjà largement effectif, est celui d'une embauche au sein des entreprises qui se sont ralliées à des degrés divers au modèle du logiciel libre; ces embauches se traduisent le plus souvent par une allocation du temps de travail de l'individu entre, d'une part des tâches dévolues aux objectifs propres de l'entreprise et, d'autre part, une implication dans les activités coopératives du bazar. Le second mode de compensation relève du secteur public et doit être considéré comme l'une des manières d'intégrer le soutien au logiciel libre dans les objectifs de politique technologique. Il pourrait s'agir d'un principe de rémunération bénéficiant à des développeurs confirmés, impliqués pour partie de leur temps dans des projets publics de recherche.

5. UN MODÈLE DE COALITION

De manière très stylisée, soit \mathcal{I} la population des utilisateurs du libre, de cardinal N , lesquels peuvent faire le choix individuel de participer ou non aux efforts collectifs de développement. On considère que ces efforts collectifs engendrent, pour chacun des N individus de \mathcal{I} , une externalité d'usage, résultant des améliorations apportées à la base des logiciels. Nous représentons cette externalité sous la forme d'une fonction $\phi(n)$ du nombre d'individus coopérant. $\phi(n)$ est bien entendu croissante et on admettra facilement que, du fait des difficultés de coordination qui croissent avec le nombre d'individus impliqués, $\phi(n)$ est strictement concave.

De manière très simplifiée, on suppose, dans un premier temps, que le coût individuel de participation, par unité de temps, est uniforme et désigné par CP , de même que pour les effets espérés en termes d'apprentissage et de réputation, désignés par KR . On peut alors écrire le gain net, à chaque période de temps, pour un développeur individuel i , selon qu'il coopère ou non:

$$\begin{aligned} G_c &= \phi(n) + KR - CP && \text{s'il coopère} \\ G_d &= \phi(n) && \text{s'il ne coopère pas} \end{aligned}$$

Un sous-ensemble \mathcal{J} de \mathcal{I} , de cardinal n^* , constitue une coalition stable, sous la double condition de stabilité interne:

$$\phi(n^*) - \phi(n^*-1) + KR - CP > 0$$

stabilité externe:

$$\phi(n^*+1) - \phi(n^*) + KR - CP < 0$$

soit donc que

$$\Delta\phi(n^*) > CP - KR > \Delta\phi(n^*+1)$$

Un équilibre existe donc, du fait de la concavité de ϕ , sous la seule condition d'une consistance suffisante des effets d'apprentissage et réputation KR vis-à-vis du coût CP de la coopération. Dans de telles conditions, n^* constitue une taille limite d'efficacité et, si N est suffisamment grand, il peut devenir nécessaire de structurer le développement du libre autour de projets qui rassemblent des communautés de développeurs-utilisateurs partiellement spécialisées, quoique non-imperméables les unes aux autres.

Considérons à présent l'élargissement contemporain du logiciel libre en introduisant, dans l'expression des gains individuels, une externalité négative, décrite comme une fonction $FR(N)$ croissante et concave du nombre N des utilisateurs. On suppose en outre que les effets d'apprentissage et de réputation $KR(i)$ espérés par un individu i , en retour de sa participation à l'effort collectif, sont l'objet d'une certaine distribution sur la population des développeurs.

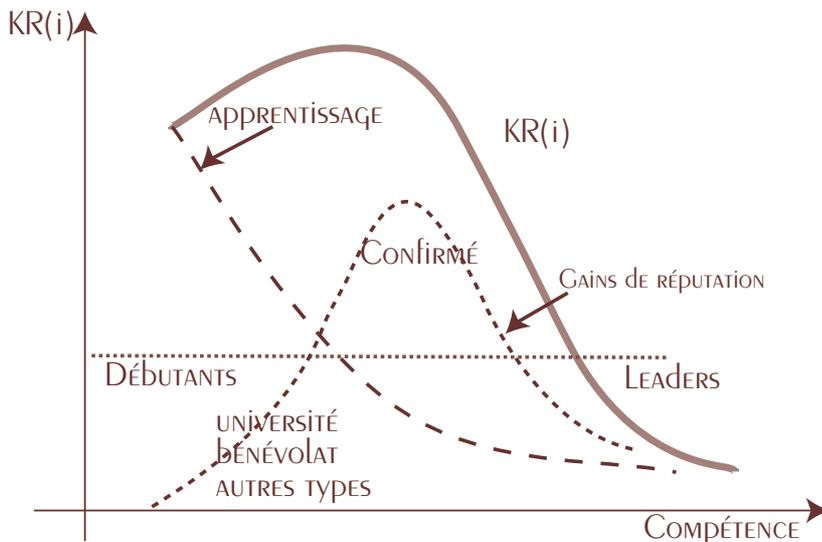
Pour un développeur, l'expression générale du gain relatif, entre coopération et non-coopération, peut alors s'écrire

$$\Delta g(i) = \Delta\phi(n) + KR(i) - FR(N) - CP$$

Cette fois, la condition de consistance des effets incitatifs est plus difficile à réaliser du fait de la présence de cet effet de frustration $FR(N)$. Si l'on considère une sous-population \mathcal{J} de taille n^* , coalition stable réalisable en l'absence de l'externalité négative, l'introduction de cette dernière va entraîner dans le camp de la défection une fraction

$\eta(N)$ de la population, correspondant aux individus dotés des $KR(i)$ les plus faibles et pour lesquels le gain relatif $\Delta g(i)$ devient négatif. Ces défections ont elles mêmes une répercussion à la baisse sur l'externalité positive $\Delta\phi(n - \eta(N))$ et le système peut alors évoluer vers un nouveau point fixe $n' > 0$ qui constituerait un nouvel équilibre, de moindre efficacité ou même s'effondrer hors de toute coalition réalisable, si N est suffisamment grand.

Le profil de la distribution des $KR(i)$ joue bien entendu un rôle déterminant dans la dynamique de recombinaison ou d'effondrement du système. Les premiers individus incités à faire défection sont ceux qui sont le moins en attente d'effets d'apprentissage et de réputation. Or ces effets varient de manière importante, avec le niveau de compétence atteint par l'individu. Ainsi on peut attendre qu'un individu proche de ses débuts compensera de faibles gains en réputation par d'importants effets d'apprentissage, avant d'entamer une progressive construction de sa réputation qui s'appuieront largement sur les compétences acquises. En revanche, un individu qui bénéficie d'une certaine reconnaissance acquise, cumulera des gains de réputation de plus en plus faibles.



Effets d'apprentissage et de réputation selon le niveau de compétence

Par conséquent, les individus les plus enclins à faire défection, lorsque l'audience des logiciels libres s'élargit, sont précisément les individus les plus compétents. Si nous cherchions à affiner les termes du modèle en tenant compte des niveaux de compétence dans la contribution à la production du bien public, ce sont clairement les individus dont l'implication est la plus efficace, qui seraient les premiers à prendre leurs distances avec le modèle du bazar. En outre, on peut aussi considérer que l'effet de frustration individuelle est d'autant plus im-

portant que l'individu est d'un niveau de compétence élevé et se trouve en position d'attendre des gains plus importants, ce qui ne peut que renforcer le phénomène. De telles considérations pourraient cependant être nuancées par la prise en compte d'un niveau d'engagement militant des individus, en particulier pour les leaders du mouvement, dont la défection supposerait l'abandon de cet objectif militant et aurait évidemment des conséquences fortes en termes de rupture de l'équilibre.

Conclusion

Ainsi, le logiciel libre constitue-t-il un modèle intéressant, à la fois parce qu'il paraît bien adapté aux caractéristiques de l'innovation dans un domaine industriel comme le logiciel, et parce qu'il est, sans aucun doute, un modèle de mutualisation des connaissances, garant de la qualité des solutions adoptées et du caractère concurrentiel des marchés.

Longtemps confiné dans un milieu restreint, le logiciel libre est entré aujourd'hui, du fait d'Internet, dans une phase de démultiplication des interactions entre développeurs et de pénétration du monde marchand. Son poids, en termes de part de marché, reste encore extrêmement réduit, mais sa popularité va grandissante et son potentiel est immense. Reste toutefois que, dans son opposition avec la logique propriétaire, le décollage du libre est contingent d'un certain nombre de facteurs qui sont avant tout institutionnels et politiques. Deux aspects nous en semblent particulièrement à retenir ici.

Le premier est relatif aux régimes de protection de la propriété intellectuelle qui seront appliqués, dans le futur, au domaine du logiciel. Le choix d'une systématisation du recours à la brevetabilité constituerait une menace redoutable pour le devenir du libre.

Le deuxième facteur est relatif à la nécessité de corriger les défauts d'incitation du modèle, en introduisant un principe de rémunération des développeurs, du moins des plus compétents d'entre eux.

Cette pratique est d'ores et déjà avaluée par les pratiques des entreprises qui se sont ralliées au monde du libre. Reste à déterminer, pour les pouvoirs publics, si elle doit à son tour compter parmi les mesures de politique technologique qui pourraient viser à renforcer la viabilité du modèle du libre. On conçoit facilement qu'une telle orientation de politique technologique ne pourrait se concevoir à la seule échelle nationale et devrait, pour le moins, viser une mise en place à l'échelle de l'Union Européenne.

Bibliographie

- Allen R.** (1983), *Collective invention*, Journal of Economic Behavior and Organization, 4, p.1-24
- Alper J.** (1999), *L'envol des logiciels libres*, La Recherche, 319, Avril, p.27-29
- Bessen J. et Maskin K.** (2000), *Sequential Innovation, Patents and Imitation*, MIT, Dept of Economics, Working Paper No00-01, January.
- Clément-Fontaine M.** (1999), *La licence Publique Générale GNU*, Mémoire de DEA
Droit des Créations Immatérielles, Université de Montpellier I, <http://crao.net/gpl/>
- Dasgupta P. et David P.A.** (1994), *Towards a new economics of science*, Research Policy 23(5), p.487-521
- David P.** (1997), *Communication norms and the collective cognitive performance of 'invisible colleges'*, in G.B. Navaretti (ed.), *Creation and transfer of knowledge: institutions and incentives*, Physica Verlag Series
- David P. et Foray D.** (1995), "Accessing and expanding the science and technology knowledge base", STIRreview, no16, OCDE, Paris, p.13-68
- Dang-Nguyen G. et Pénard T.** (2001) *Don et coopération sur Internet, Un essai d'explication économique*, Revue Economique, à paraître.
- D. Desbois, N. Jullien, T. Pénard, A. Poulain-Maubant, J. Vétois et J. B. Zimmermann** (Eds.) (1999) *Logiciels Libres: de l'utopie au marché*, L'Harmattan/Terminal .
- Foray D. et Hilaire Perez L.** (2000), *The economics of open technology: collective organization and individual claims in the "fabrique lyonnaise" during the old regime*, Conference in honor of Paul David, Turin, Mai 2000
- Ganagé M. et Bonnet R.** (1998), *Linux: l'explosion du phénomène des logiciels libres*, L'Ordinateur Individuel, No 97, Juillet-Août.
- Hart R., Holmes P. et Reid J.** (2000), *The Economic Impact of Patentability of Computer Programs*, Report to the European Commission, Study Contract ETD/99/B5-3000/E/106.
- Heraud J.A.** (1995), *Brevet et contexte institutionnel de la création technologique*, in Baslé M., Dufourt D., Héraud J.A. et Perrin J., *Changement institutionnel et changement technique*, CNRS Editions, Paris
- Lakhani K. et von Hippel E.** (2000), *How open source software works: Free user-to-user assistance*, MIT Sloan School of Management, Working Paper #4117
- Lerner J. et Tirole J.** (2000), *The simple Economics of Open Source*, Harvard Business School et Institut d'Économie Industrielle
- Mimeo Lucas A.** (1987), *Le droit de l'informatique*, Thémis, PUF
- Quah** (1999), *The weightless economy in economic development*, London School of Economics, Economic Department, Londres
- Raymond E.S.** (1998), *La Cathédrale et le Bazar*, traduit par Blondeel S., http://www.liff.fr/~blondeel/traduc/Cathedral-bazaar/Main_file.html
- Scotchmer S.** (1991), *Standing on the shoulders of giants: cumulative research and the patents law*, Journal of Economic Perspectives, vol. 5, no 1, hiver, pp. 29-41
- Smets-Solanes J. P. et Faucon B.** (1999), *Logiciels libre - Liberté, Égalité, Business*, Edispher, Paris
- Vivant M.** (1993), *Une épreuve de vérité pour les droits de propriété intellectuelle: le développement de l'informatique*, in *L'avenir de la propriété intellectuelle*, Librairies Techniques, collection le Droit des Affaires, Paris
- Zimmermann J. B.** (1995), *L'industrie du logiciel: de la protection à la normalisation*, in Baslé M. et al. (1995)
- Zimmermann J. B.** (1998), *L'industrie du logiciel: ébauche d'une approche prospective*, Terminal No 75, Hiver 97 - Printemps 98. ■

CONCOURS DE LA MEILLEURE NOUVELLE

Cette année encore, le flash informatique avait invité les étudiants et le personnel des universités et écoles polytechniques suisses à participer au concours de la meilleure nouvelle. Peu nombreuses ont été les contributions, mais heureusement, elles étaient dans l'ensemble de bonne qualité. La nouvelle **Denrée périssable** a valu à son auteur **Adrien Bürki** le premier prix d'une valeur de frs. 1000.- offert par la librairie Ellipse de Genève , remis lors de la rencontre-débat **les logiciels libres** à l'EPFL qui a eu lieu le 12 juin dernier.



Le jury a retenu en plus du texte gagnant, deux autres nouvelles que vous pourrez lire sur le Web à l'adresse: <http://sic.epfl.ch/publications/FI01/fi-sp-01/>; ce sont:

Pingouins sur banquise, Aymar

Le Pingouin, GNU Generation

Le jury remercie les participants au concours et espère que les auteurs potentiels se mobiliseront encore plus nombreux l'année prochaine.

Les membres du Jury



DENRÉE PÉRISSABLE

PAR ADRIEN BÜRKI, ÉTUDIANT EN LETTRES, UNIVERSITÉ DE LAUSANNE

« *Il faut être absolument moderne* »

Arthur Rimbaud

ET puis il l'aperçut, tournant le coin de la rue à la suite du criant caddie de supermarché qu'il poussait, comme chaque jour encombré de l'immense carton mal ficelé qui laissait entrevoir son contenu, le reflet sur la grise vitre bombée. Le clochard, le bras gauche caché sous la doublure de son manteau, laissait pendre à son côté la manche inutile. La première fois qu'il l'avait vu, à ce même carrefour, Adrien l'avait pris pour un manchot, et n'avait pu réprimer un sourire intérieur en constatant la démarche nerveuse du bonhomme, balançant le buste alternativement de gauche et de droite à chaque pas, les pointes des pieds écartées, qui l'apparentait irrésistiblement à un grand oiseau glissant maladroitement sur le verglas, à un pingouin.

Adrien s'avança alors, et le clochard vit venir à lui un homme jeune et chiquement vêtu qui pouvait être tout aussi bien cadre supérieur, employé de banque, publicitaire, ou alors autre chose, comment eût-il pu deviner? Comme les deux jours précédents, l'individu déposa dans le bol son obole, puis demeura sur place, immobile. Le mendiant attendit lui aussi, les yeux baissés, puis voyant que l'autre ne s'en allait pas, ébaucha de la tête un remerciement; vain: l'homme encravaté ne bougea pas. Le vieux finit par jeter un regard au visage perché haut au-dessus de lui, qui était maintenant tourné vers le gros carton entrouvert.

- On peut savoir ce que vous voulez?
- C'est un ordinateur que vous avez là, n'est-ce pas?
- Tout juste.
- Un bel ordinateur...
- L'est pas à vendre.
- Je ne veux pas l'acheter. Mais j'avoue ma curiosité: je voudrais savoir...

- ... ce que fabrique un vieux clodo avec une bécanne hyper-moderne, hein?
- Le type acquiesça en silence.
- C'est pour ça, hein, l'aumône?! Un marché: une pièce contre une histoire. Vous aimez ça, les histoires de pauvres?
- Adrien se pencha vers l'haleine avinée.
- Ce qui me fascine, me passionne, c'est la déchéance. Savoir comment on devient ce que vous êtes. Je suis écrivain.
- Vous avez de drôles de passions, monsieur l'Écrivain. Bah! Vous l'aurez, votre histoire, rassurez-vous! Je ne sais pas si elle est à votre goût, ni si elle est croyable ou incroyable; peu importe: elle est la mienne.

Vous connaissez ce magasin, Morby Electronique, près de la place Maurice? C'est vrai qu'on ne le remarque guère, il donne sur une ruelle peu fréquentée. Je l'ai découvert par hasard, un jour où j'étais jeune, plus jeune que vous, et promis à un riche avenir. Or j'étais à la recherche d'un ordinateur performant; j'y suis entré.

C'était une boutique exiguë, qui avait dû auparavant être une sorte de cave, car les murs étaient faits de pierres apparentes. Mais là s'arrêtait le côté vieillot de l'endroit: car partout, sur des étagères, des présentoirs, s'offrait au regard tout le dernier cri de l'informatique, un appareillage complet et flambant neuf, miroitant.

- Vous allez trouver votre bonheur, prononça une voix derrière moi, sans que je pusse discerner s'il s'agissait d'une question ou d'une affirmation péremptoire.

Je me suis retourné: c'était le vendeur. Un homme banal, quelconque, prématurément vieilli, qui a su me conseiller le matériel adéquat pour mon usage personnel. Bien sûr, jeune comme j'étais, j'ai insisté pour obtenir ce qui se faisait de plus moderne. Alors, d'une voix malicieuse, il m'a lancé cette sentence: « Mais aujourd'hui, un ordi-

nateur est démodé dès qu'on a franchi avec lui la porte du magasin! » Nous avons ri de cette boutade, échangé quelques mots, puis je l'ai payé en espèces, et j'ai emporté mon ordinateur chez moi.

L'installation fonctionnait à merveille. Fier de ma nouvelle acquisition, j'ai invité mes amis et mes collègues à venir admirer l'engin et ses insoupçon-



nables capacités. Certains faisaient mine de s'extasier, d'autres posaient des questions techniques, mais il me semblait

que, s'ils s'amusaient de mon naïf enthousiasme, ils ne partageaient pas pour autant ma conviction de bénéficier des attraits de la plus récente modernité. Déjà on me suggérait des extensions, des cartes graphiques et d'autres options dont je feignais alors d'avoir envisagé l'achat. Mais intérieurement je savais que ce que j'avais me suffisait amplement et que je ne succomberais pas aux sirènes technologiques.

J'ai tenu trois bonnes semaines.

Quand je suis retourné à Morby Electronique, le vendeur m'a reconnu et m'a salué d'un air entendu.

– Alors? Qu'est-ce qu'il vous faut de plus, jeune homme?

Résultat, je suis ressorti du magasin muni de toutes sortes d'accessoires garantis extrêmement utiles pour une utilisation optimale. En refermant la porte, faisant retentir la clochette, j'ai repensé en souriant à la maxime du vendeur, décidément terriblement vraie.

Ensuite, la vie a continué normalement, entre mon travail et mes amis; à deux ou trois reprises j'ai complété mon lot d'options à la boutique d'informatique. Mon appartement vrombissait comme un réacteur d'avion, ma chambre s'encombrait de câbles et de voyants verts et de bruits exotiques.

C'est à peu près à ce moment que j'ai remarqué que le regard de mes collègues commençait imperceptiblement à changer, celui des femmes surtout, comme si elles évitaient de me regarder. Des amis, avec un tact maladroit, s'enquerraient de ma

santé, m'assuraient de leur discrétion totale si j'avais quelque confiance à leur faire. Mais les seuls problèmes qui me travaillaient étaient des problèmes financiers: je n'occupais alors dans l'entreprise qu'une place subalterne, assurément provisoire, mais mon salaire actuel suffisait tout juste à couvrir mes dépenses de disques durs, de moniteurs grand format, de choses, de trucs et de machins à puces. Peut-être après tout avaient-ils raison: ces derniers temps, obnubilé par mon ordinateur, j'avais passablement négligé de prendre soin de moi-même. J'ai passé ma main sur mon menton: cela râpait.

Rasé de frais et chaudement douché, je me suis examiné avec satisfaction dans la glace embuée. Et c'est là que je l'ai découvert.

– Qui ça? demanda Adrien.

Fatigué d'être debout, il s'était assis auprès du vieillard, et regardait dans la même direction que lui, de l'autre côté de la rue.

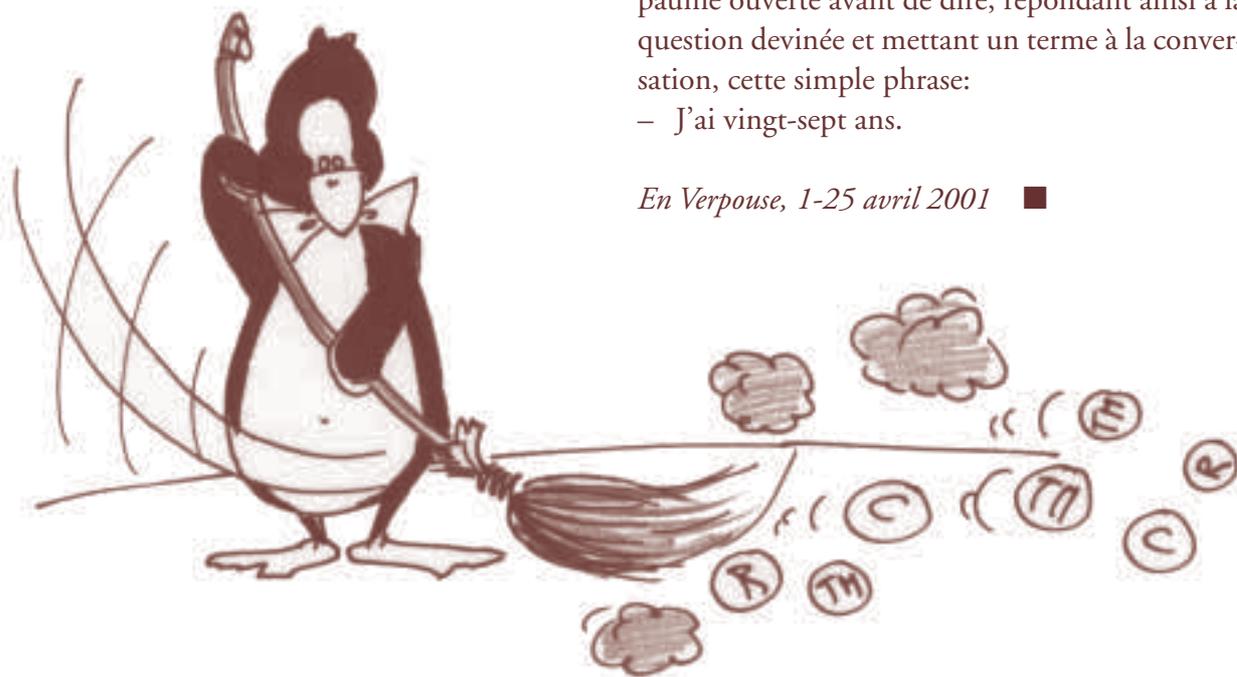
Les passants passaient sans leur prêter attention. L'écrivain ne pouvait discerner clairement les visages des gens, trop haut au-dessus de lui; le défilé incessant de pieds et de jambes, voilà tout ce qu'il pouvait distinguer. Il reporta les yeux sur le clochard; de profil, son nez pointu et rougi semblait le bec d'un quelconque palmipède, un pingouin ou un toucan. Mais le toucan n'est pas un palmipède, ou bien? Il éluda la question, car son compagnon avait déjà répondu:

– Le premier cheveu blanc. Vous pensez si je l'ai arraché! Et puis je n'y ai plus pensé. A quoi bon me faire du souci alors que mes amis s'en faisaient pour moi! Enfin, seulement ceux qui me restaient fidèles, car certains se sont peu à peu détachés de moi. Mais les autres paraissaient pleins de sollicitude, m'invitaient à dîner chez eux, retenaient la porte de l'ascenseur, se proposaient d'aller me chercher un café. J'étais naturellement gêné d'être l'objet de ces prévenances, mais aussi soulagé, car je me sentais alors souvent fatigué - le manque de sommeil, pensais-je - et j'étais content d'éviter d'avoir à gravir une longue rampe d'escalier ou à passer des heures dans un grand magasin à pousser un charriot.

J'ai pris une semaine de vacances, qui m'a permis de me reposer chez moi et d'évacuer les soucis du travail. J'en ai profité pour ajouter à mon ordinateur deux ou trois logiciels indispensables - le vendeur m'a cligné de l'œil en me répétant sa désormais fameuse phrase « Un ordinateur est démodé dès qu'on... », mais je suis parti avant d'entendre la fin de cette blague pour moi éculée. Mais le dimanche soir, après un long bain brûlant - goût que je n'avais que depuis peu - j'ai commis l'erreur de me scruter scrupuleusement dans le miroir, et j'y ai vu que mon congé ne m'avait pas - bien au contraire! - été profitable: j'avais le front strié de lignes de souci, et au coin des yeux l'ébauche d'un réseau de ridules finissant au bas de mes tempes. Il fallait sur-le-champ que je cesse de me négliger à ce point, sous peine de risquer, qu'en savais-je à ce moment! une dépression nerveuse, ou quelque chose dans ce genre-là.

Sur mon bureau m'attendait une convocation chez mon patron. Restructurations... Regrettable situation... Votre profil ne... Bref, viré, sans le sou, épuisé, l'ombre de moi-même, la fin de l'histoire s'impose tout naturellement.

- D'accord, dit Adrien, mais je ne comprends pas: cette histoire doit dater au moins de quarante ou cinquante ans, tandis que les ordinateurs...
- Oh, ça suffit maintenant! J'ai eu mon franc, vous avez eu votre récit, nous sommes quittes!



Il se tut, et l'écrivain de son côté. Ni l'un ni l'autre n'osaient se lever et partir, retenus tous les deux par un fil invisible. Et finalement le vieux reprit:

- Un mois plus tard, criblé de dettes et de douleurs, voûté, méconnaissable, je me suis rendu au magasin d'informatique. Le vendeur a eu en me voyant un sourire triste. « Il faut que je vous dise la vérité, m'a-t-il dit. Ma petite plaisanterie, vous vous souvenez? eh bien ce n'est pas vraiment, et même pas du tout une plaisanterie: lorsque vous quittez mon magasin, votre ordinateur était certes démodé, mais ce n'était pas lui qui devenait subitement archaïque, comme on aurait pu le croire, par rapport à vous et à votre présent; au contraire, c'était le présent, incluant l'ordinateur, qui reculait d'un coup dans le passé, vous laissant à chaque achat un peu plus vieux. Je suis désolé, mais je n'y peux rien; le monde est ainsi fait. »

Les yeux bleus du vieillard luisaient de larmes lorsqu'il tourna son visage flétri vers celui de son interlocuteur. Leur éclat éteint le suppliait de s'en aller.

Adrien se leva sans oser poser la question qui le dévorait. Il tendit la main au clochard pour qu'il la serre, mais le vieux, sans cesser de fixer son regard sur celui de l'écrivain, farfouilla dans son récipient et en retira une pièce qu'il déposa dans la paume ouverte avant de dire, répondant ainsi à la question devinée et mettant un terme à la conversation, cette simple phrase:

- J'ai vingt-sept ans.

En Verpouse, 1-25 avril 2001 ■

DATA MINING AVEC R DANS UN MONDE LIBRE



DIEGO.KUONEN@epfl.ch ET REINHARD.FURRER@epfl.ch,
EPFL, DÉPARTEMENT DE MATHÉMATIQUES



R est un langage et un environnement pour les calculs statistiques et leurs représentations graphiques. **R** est similaire au système **S** qui a été récompensé par le *Software System Award* de l'ACM (*Association for Computing Machinery*) et qui est la plate-forme du logiciel commercial S-Plus. Rappelons quelques technologies reconnues par la récompense ACM à savoir UNIX, TeX, PostScript, TCP/IP, World-Wide Web, Tcl/Tk, et Apache. La citation de l'ACM contenait la phrase suivante: «... *system, which has forever altered how people analyze, visualize, and manipulate data.*»

Le but de l'article précédemment paru dans le FI 2/01 [1] était de fournir un point de départ pour les novices intéressés par **R** [2]. Fin juin 2001, la version 1.3.0 de **R** est sortie, contenant un portage stable pour MacOS/X [3]. **R** est un exemple parmi tant d'autres du succès incontestable des logiciels libres – les éléments essentiels d'un monde libre.

Le présent article se propose d'illustrer les interfaces existantes entre **R** et des bases de données relationnelles, ces interfaces étant un premier pas des logiciels statistiques modernes, comme **R**, vers la reconquête par les statisticiens du domaine du *Knowledge Discovery in Databases* (KDD). Ce propos est illustré à l'aide de pingouins, enfermés derrière des fenêtres d'un zoo et qui aimeraient partir vers un monde libre.

QU'EST-CE QUE LE DATA MINING?

«*We are drowning in information, but starving for knowledge.*» (John Naisbett dans son livre *Megatrends*)

De manière générale, on peut définir le *Data Mining* (ou *Exploitation des Gisements de Données*) comme l'extraction, à partir de gros volumes de données, d'informations ou de connaissances originales, auparavant inconnues, potentiellement utiles.

Selon *SAS Institute Inc.*, il s'agit du processus de sélection, exploration, modification et modé-

lisation de grandes bases de données afin de découvrir des relations entre les données jusqu'alors inconnues.

Le *data mining* correspond donc à l'ensemble des techniques et des méthodes qui, à partir de gros volumes de données, permettent d'obtenir des connaissances exploitables. Son utilité est grande dès lors qu'une entreprise possède un grand nombre d'informations stockées sous forme de bases de données.

Il existe une distinction précise entre le concept de KDD (*Knowledge Discovery in Databases* ou *Découverte de Connaissances dans les Bases de Données*) et celui de *data mining*. En effet, ce dernier n'est que l'une des étapes du processus de découverte de connaissances correspondant à l'extraction des connaissances à partir des données. Pour pouvoir réaliser des études de *data mining* il faut d'abord disposer d'un *Data Warehouse* (*Entrepôt de Données*).



Les applications du *data mining* sont multiples: la grande distribution, la vente par correspondance, les opérateurs de télécommunications, les banques et les assurances, l'étude des génomes dans la bioinformatique, par exemple, en trouvant des gènes dans des séquences d'ADN, etc. Le domaine majeur où le *data mining* a prouvé son efficacité est la gestion de la relation client, CRM (*Customer Relationship Management*). En effet, dans ce cas le *data mining* permet d'accroître les ventes par une meilleure connaissance de la clientèle. Une bonne référence sur ce dernier point est le livre de Berry et Linoff [4]. De nombreuses adresses Web concernant le *data mining* sont disponibles sur le site Statoo.com [5] et sur celui du premier auteur [6].

Le *data mining* et ses outils, bien qu'utilisant la démarche et des techniques statistiques, sont appelés à être utilisés par des non-statisticiens. Regardons pourquoi.

Le domaine des statistiques, spécialement des statistiques dites mathématiques, a commencé à croître il y a environ 100 ans. A l'époque les jeux de données étaient de petite taille et les calculs s'effectuaient manuellement. L'ère des statistiques ma-

thématiques s'est terminée le jour où, grâce aux ordinateurs, les statistiques dites computationnelles sont apparues. Cependant, le stockage électronique des données restait toujours coûteux et les volumes des données toujours limités. Et, comme les statistiques font traditionnellement partie du domaine des mathématiques et non pas de l'informatique, la statistique trouvait son intérêt premier dans la théorie mathématique sous-jacente, et non pas dans les aspects du calcul et du stockage de données à traiter. Ceci explique pourquoi les logiciels statistiques étaient habituellement bons en lecture, respectivement en importation de données depuis un fichier, mais ont malheureusement ignoré pendant trop longtemps le fait que des données reposaient dans des bases de données et l'intérêt qu'il y aurait de s'interfacer avec ces bases de données. Depuis le début des années 1990 le nombre de très grandes bases de données, souvent distribuées, n'a pas arrêté de croître. Les agences gouvernementales, les grands détaillants ou les commerçants en ligne font face maintenant aux bases de données énormes, difficiles à analyser au delà de simples indicateurs sommaires. Ainsi, les informaticiens ont tiré profit de ce manque de connaissance de base de données des statisticiens et *ont inventé* une discipline appelée KDD. La question reste posée: cette nouvelle discipline est-elle en concurrence avec les statistiques?

Puisque KDD a pour but de recréer et de veiller et données, d'autoriser la conception expérimentale, et l'analyse des données en utilisant les modèles de découverte, de visualisation, de faire des prévisions et des classifications, elle se situe clairement dans le domaine des statistiques. En résumé, on peut dire comme Daryl

Pregibon [7]: *KDD = Statistics at Scale and Speed*. Mais, il reste encore à la statistique de relever les défis du *data mining*.

Le processus usuel du *data mining* est représenté dans la Figure 1 et peut se résumer ainsi:

1. Définir une requête pour extraire toutes les données souhaitées de la base de données;
2. Exporter les données sélectionnées dans un fichier;
3. Si nécessaire, les convertir dans un format qui facilite l'importation dans un logiciel statistique tel que **R**;
4. Importer les données dans ce logiciel;
5. Analyser les données dans ce logiciel.

Il faut recommencer le processus tant que l'on n'a pas extrait toutes les données appropriées ou bien si les données fondamentales dans la base de données changent, ce qui nécessite la mise à jour des résultats.

La possibilité de manipuler des données se trouvant dans des bases de données est très importante, mais elle manque dans beaucoup de logiciels statistiques. Travailler directement sur des bases de données avec des outils interactifs semble être beaucoup plus difficile que de travailler sur des données structurées spécialement pour des applications statistiques. Mais c'est la seule possibilité qui s'offre aux statisticiens pour regagner le terrain de la KDD occupé actuellement par les informaticiens.

A ce titre le futur de **R** est prometteur en visant à combler cette lacune. Le premier pas a été fait en mettant à disposition des interfaces avec des bases de données relationnelles.

LES BASES DE DONNÉES ET SQL

Les bases de données sont devenues une partie intégrante de notre vie. Sans elles certaines tâches deviendraient très pénibles ou même impossibles. Les banques, les universités et les bibliothèques sont trois exemples d'organismes qui dépendent fortement de bases de données. Sur Internet, les moteurs de recherche, les achats en ligne, et même les serveurs de noms (DNS)

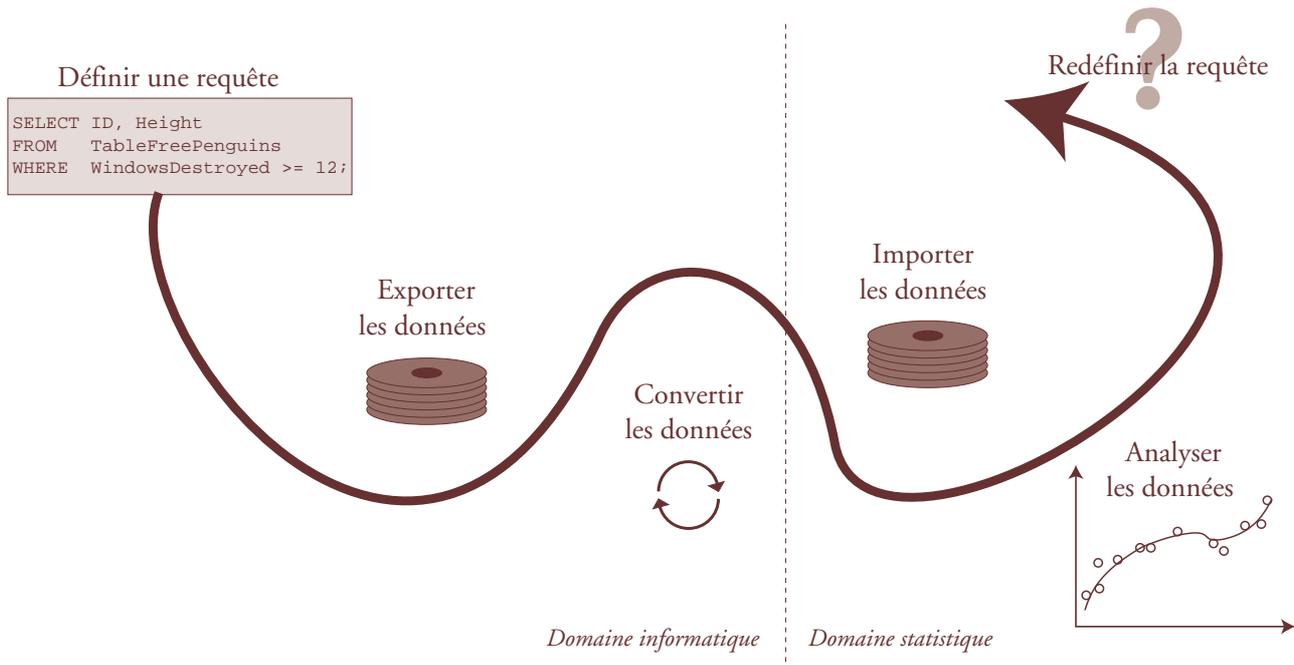


FIGURE 1: LE PROCESSUS USUEL DU DATA MINING.

seraient impossibles à réaliser sans l'utilisation d'une base de données.

On nomme souvent serveur de base de données l'ordinateur sur lequel une ou plusieurs bases de données sont rendues accessibles. La connectivité à ces bases de données est une partie de plus en plus importante du calcul statistique et les interfaces aux bases de données volumineuses deviennent importantes pour les applications d'extraction de données.

Il y a plusieurs types de DBMS ou SGBD (*Data Base Management Systems* ou *Système de Gestion de Base de Données*):

- les fichiers textes et csv;
- les tableurs (programmes et formats);
- les bases de données de fichier *plats* comme DBase;
- les bases de données hiérarchiques (telles que HDF5 [8]);
- les bases de données relationnelles *maigres* comme Access, MySQL [9, 10, 11], MiniSQL (également connu sous le nom de mSQL, [12]); et
- les bases de données relationnelles *lourdes* comme Oracle, DB/2, Microsoft SQL Server, Informix, Sybase, PostgreSQL [13].

Parfois l'utilisateur a le choix du DBMS. Plus souvent il doit utiliser une base de données existante ce qui ne lui laisse pas le choix. La plupart des DBMS relationnels sont des systèmes du type client/serveur, et beaucoup permettent la transmission par TCP/IP. La plupart des DBMS viennent avec un moniteur, un client basé texte, certains ont des clients GUI, et presque tous ont une API C ou C++. Notons que seul le DBMS peut accéder à la base de données par l'intermédiaire de commandes, qui sont normalement introduites avec un dialecte de SQL (*Structured Query Language* ou *Langage de Requête Structuré*).

SQL est le principal langage informatique de création et de manipulation de bases de données relationnelles, permettant tant la définition que la manipulation et le contrôle d'une base de données relationnelles. SQL est défini par des normes et est implanté dans les systèmes de gestion de base de données de la plupart des fournisseurs, parfois avec des modifications mineures dans la syntaxe. SQL possède deux caractéristiques fondamentales: il est ensembliste et non-procédural. La première caractéristique veut dire qu'il porte toujours sur des ensembles (des tables), et qu'il produit toujours des ensembles (des tables). La seconde veut dire que le langage ne décrit pas un algorithme à exécuter afin d'obtenir le résultat désiré, comme les langages de programmation traditionnels (FORTRAN, C), mais qu'il décrit uniquement le résultat désiré, sans se préoccuper de la manière dont le système de gestion de base

de données va exécuter la commande. SQL est un outil très puissant et facile à utiliser, mais qui requiert qu'on ait constamment à l'esprit le fait qu'on travaille sur des ensembles. En particulier, il faut se souvenir qu'une jointure fait toujours le produit de deux ou de plusieurs tables et qu'il faut donc, la plupart du temps, restreindre la jointure aux rangées désirées en indiquant une condition à remplir.

Plus généralement, SQL contient trois types de langages:

- un langage de requêtes;
- un langage de manipulation de données;
- un langage de définition de données.

Voici un petit exemple pour illustrer les principes fondamentaux de SQL. Dans une base de données relationnelles, les données sont stockées dans des tables. Le tableau suivant, TableFreePenguins, contient des données sur des pingouins vivant dans un zoo derrière des fenêtres. Ces pingouins aimeraient s'échapper vers un monde libre. Pour ce faire, ils doivent casser les fenêtres imposées par le fabricant du zoo. Le tableau TableFreePenguins met en relation le numéro d'identification des pingouins (*ID*), leur race (*Species*), leur taille en cm (*Height*), leur poids en kg (*Weight*) et le nombre de fenêtres cassées (*WindowsDestroyed*):

ID	Species	Height	Weight	Windows Destroyed
21	Aptenodytes patagonicus	94	14	98
01	Aptenodytes forsteri	112	30	2000
13	Pygoscelis adeliae	50	4	4
24	Pygoscelis papu	65	6	95
55	Pygoscelis antarctica	47	4	31
76	Eudyptes chrysocome	44	2	1

Maintenant, supposons que nous voulions obtenir la taille et le poids de chaque pingouin. On utilisera l'instruction SELECT, comme ceci:

```
SELECT Height, Weight
FROM TableFreePenguins;
```

Voici le résultat de l'interrogation de notre base de données:

Height	Weight
94	14
112	30
50	4
65	6
47	4
44	2

Pour obtenir toutes les colonnes d'une table sans avoir à taper tous les noms de colonne, on peut utiliser:

```
SELECT * FROM TableFreePenguins;
```

La façon de saisir les instructions SQL et d'accéder à la base de données varie en général d'un DBMS à l'autre; consultez un gourou prêt à vous aider si vous devez utiliser une de ces bases de données.

Il y a sept opérateurs relationnels en SQL: «=», «<>» ou «! =», «<», «>», «<=», «>=». La clause WHERE est utilisée pour spécifier que l'on affiche seulement certaines lignes de la table, selon un critère défini par cette clause WHERE. De façon plus claire, prenons l'exemple suivant où l'on désire savoir l'*ID* et la taille des pingouins ayant cassé au moins 98 fenêtres:

```
SELECT ID, Height
FROM TableFreePenguins
WHERE WindowsDestroyed >= 98;
```

On aura l'affichage:

ID	Height
21	94
01	112

La description de `WHERE, WindowsDestroyed >= 98`, est appelée une condition. On voit que ce sont les pingouins les plus grands qui ont cassé le plus grand nombre de fenêtres. Le monde libre n'est donc pas uniquement désiré par les petits.

Pour ajouter une difficulté, nous voulons aussi que cette liste soit classée par ordre alphabétique selon la race du pingouin. Pour cela, nous utiliserons la clause `ORDER BY`:

```
SELECT ID, Height
FROM TableFreePenguins
WHERE WindowsDestroyed >= 98
ORDER BY Species;
```

Ce qui donnera

ID	Height
01	112
21	94

Le but de cet article n'est pas de vous donner une introduction à SQL, que vous trouverez sur de nombreux sites Web comme [14], dans le manuel et les HOWTOs de votre système de gestion de bases de données ou dans des livres comme [11].

R ET LES BASES DE DONNÉES

Actuellement, il y a sur le «Comprehensive **R** Archive Network» (CRAN, [3]) trois modules d'interface de **R** avec des DBMS:

- RPgSQL pour PostgreSQL;
- RMySQL pour MySQL;
- RmSQL pour MiniSQL;

utilisant des interfaces en C. Celles-ci et RODBC sont décrits dans le manuel ***R** Data Import/Export* [15]. L'option la plus portable est de loin RODBC et, à moins que vous puissiez choisir votre système de gestion de base de données, vous n'avez probablement aucun autre choix. La *Open Database Connectivity* (ODBC) fournit une interface client commune à presque tous les DBMS populaires. Pour utiliser ODBC vous avez besoin d'un gestionnaire de pilotes pour votre OS et un pilote pour votre système de gestion de base de données. Heureusement les pilotes nécessaires sont largement disponibles, mais pas

toujours conformément aux versions récentes de l'ODBC. Les outils de base de RODBC doivent transférer des tableaux de données vers et depuis un système de gestion de base de données.

Prélever des données d'un DBMS est la tâche commune à tous. La table d'un DBMS relationnel est un concept semblable à un tableau de données (*data frame*) dans **R**. Ainsi, une idée naturelle est de créer une table contenant les données souhaitées, et de la stocker dans un data frame. RmSQL fait ceci ligne par ligne. Pour charger un module dans **R**, il faut utiliser la commande:

```
> library(«Nom du module»)
```

Par exemple, pour charger l'interface RPgSQL taper,

```
> library(RPgSQL)
```

Ainsi, on a pour RPgSQL,

```
> db.connect(dbname=»freeworld»)
> db.read.table(«TableFreePenguins»)
> db.disconnect()
```

pour RODBC,

```
> channel <- odbcConnect(«freeworld»,uid=»linus»)
> sqlFetch(channel, «TableFreePenguins», rownames =
TRUE)
> odbcClose(channel)
```

et pour RMySQL,

```
> con <- dbConnect(MySQL(), dbname = «freeworld»)
> getTable(con, «TableFreePenguins»)
> close(con)
```

Malheureusement, divers problèmes peuvent surgir:

1. Peu de DBMS font par défaut la distinction entre minuscules et majuscules. Ceci dépend aussi de l'OS. MySQL, par exemple est *case-sensitive* sous Linux, mais pas sous Windows.
2. Les DBMS ont en principe les tables non triées, il faut disposer d'un champ d'identification (clé primaire) pour pouvoir le faire.
3. Faire correspondre les types de données dans la base aux types de données de **R** peut être problématique.
4. Les valeurs manquantes sont généralement représentées dans SQL par la valeur NULL. Il faut bien s'assurer de la façon dont elles sont transférées dans **R**.
5. La lecture de la table entière d'un seul coup pourrait être une trop lourde charge pour **R**. Il faudrait donc lire bloc par bloc, et pour ceci il faut probablement utiliser SQL, ce qui peut s'avérer compliqué.

Pour quelques applications, ajouter les données provenant de **R** à une base de données est une tâche importante. Les fonctions nécessaires sont disponibles dans la plupart des interfaces de **R**.

Pour RPgSQL:

```
> data(TableFreePenguins)
> freepenguins <- TableFreePenguins
> db.connect(dbname=»freeworld»)
> db.write.table(freepenguins,write.row.names =
TRUE)
> db.disconnect()
```

pour RODBC,

```
> channel <- odbcConnect(«freeworld»,uid=»linus»)
> sqlSave(channel, freepenguins, rownames = TRUE)
> odbcClose(channel)
```

et pour RMySQL,

```
> con <- dbConnect(MySQL(), dbname = «freeworld»)
> assignTable(con, «freepenguins»,
              TableFreePenguins, overwrite=TRUE)
> close(con)
```

Nous devons nous assurer que les noms des lignes (*rownames*) sont sauvés. Une alternative est d'écrire les données dans un fichier externe et de charger celui-ci par l'intermédiaire de SQL, de l'interface ou du programme de monitoring.

Toutes les interfaces permettent aux commandes de SQL d'être envoyées au système de gestion de bases de données.

Par exemple, pour RPgSQL,

```
> db.connect(dbname=»freeworld»)
> db.execute(«SELECT ID, Height FROM freepenguins»,
            «WHERE WindowsDestroyed >= 98 ORDER BY Species»,
            clear=FALSE)
> db.fetch.result()
> db.disconnect()
```

pour RODBC,

```
> channel <- odbcConnect(«freeworld», uid=»linus»,
                       case=»tolower»)
> sqlQuery(channel, «select id, height from
                  freepenguins where windowsdestroyed >= 98 order
                  by species»)
> odbcClose(channel)
```

et pour RMySQL,

```
> con <- dbConnect(MySQL(), dbname = «freeworld»)
> quickSQL(con, «select ID, Height from
                freepenguins where WindowsDestroyed >= 98 order
                by Species»)
> close(con)
```

Le module RPgSQL fournit une interface sophistiquée à PostgreSQL et met à disposition des facilités analogues à celles décrites pour RODBC. En outre, cette interface a la notion puissante d'un *data frame* du type proxy. C'est une classe de **R** qui hérite de la classe des *data frames* usuels, mais qui prend peu d'espace car c'est une référence à une table dans PostgreSQL. Il y aura donc un avantage lorsque l'on veut accéder à de plus petites parties du tableau des données, et ceci chaque fois que les opérations d'indexation dans **R** sont traduites en requêtes SQL. Ainsi le gros de la requête va être fait dans PostgreSQL directement plutôt que dans **R**:

```
> db.connect(dbname=»freeworld»)
> bind.db.proxy(«TableFreePenguins»)
```

«TableFreePenguins» est maintenant un proxy et donc tous les accès sont fait dans la base de données directement et pas dans **R**.

```
> TableFreePenguins[, «Height»]
Height
1    94
2   112
...
> db.disconnect()
```

Tous les DBMS offrent la possibilité plus ou moins limitée d'utiliser des fonctions et des opérateurs dans des instructions SQL. Par contre, il y a un bon nombre de différences: par exemple PostgreSQL et MySQL convertissent des types de données convenablement, mais Access ne le fait pas. Tous les noms de fonctions, leur portée exacte et la possibilité de définir des fonctions par l'utilisateur diffèrent

largement. Access utilise Visual Basic pour des applications; MySQL et PostgreSQL permettent des fonctions définies par l'utilisateur. (On espère que le lecteur intéressé commence à voir les beautés d'un monde libre.) Pour chacune des trois interfaces ceci ouvre la possibilité d'inclure la fonctionnalité de **R** dans des fonctions définies par l'utilisateur, et ceci par l'intermédiaire d'une *shared library* sur Linux/Unix pour MySQL et PostgreSQL; voir aussi [16].

Dans tous les domaines de la statistique la gestion moderne des données est de nos jours basée sur des systèmes de gestion de base de données, la plupart du temps du type relationnel. Ces systèmes de gestion nous aident à organiser les données d'une manière sécurisée et accessible, même dans les environnements multi-utilisateurs. Par conséquent, le statisticien (et l'utilisateur de **R**) est confronté à de tels systèmes s'il doit extraire ou insérer des données. Il est alors souhaitable d'utiliser des méthodes simples d'accès aux données sans avoir besoin d'une expérience de pro-



grammation, l'objectif principal étant de rendre **R** indépendant du moteur de la base de données elle-même.

D'autres langages que **R** fournissent des interfaces de programmation bien définies pour des DBMS, tels que JDBC, Perl DBI et d'autres. **R** n'étant pas seulement un langage de programmation, mais également un outil pour l'analyse de données, il n'a pas seulement besoin d'une interface de programmation mais aussi d'une interface utilisateur avec DBMS. Dans ce cas on peut percevoir le système de gestion de bases de données comme mémoire. Ceci a été fait avec succès pour l'ajustement de modèles linéaires: voir [17] et [18]. Il faut toutefois tenir compte qu'actuellement des ensembles de données trop grands peuvent poser des problèmes avec **R**.

En plus du souhait d'avoir différentes interfaces pour différents moteurs, il faut s'assurer que les utilisateurs expérimentés n'ont pas de problèmes en utilisant des bases de données dans **R** avec les modules actuellement disponibles.

L'utilisation des bases de données relationnelles n'est pas si facile que ça. Compte tenu

de leurs fonctionnalités limitées, MySQL et mSQL sont les deux bases de données parmi les plus simples. D'autres comme Oracle ou Sybase nécessitent une connaissance détaillée des systèmes de gestion de bases de données. Dans un environnement client/serveur, on ne permet habituellement à un utilisateur ni de créer les bases de données lui-même, ni de les effacer. En outre, la puissance des bases de données relationnelles repose sur des relations. Il est difficile de formuler des ensembles joints dans d'autres langages que SQL.

Pour des informations sur les futures directions de **R** et de ses interfaces avec des bases de données, nous renvoyons aux articles [19] et [20] et surtout au *Omegahat Project for Statistical Computing* [16]. C'est un projet collectif dont le but est de fournir une variété de logiciels libres pour des applications statistiques. Ce projet a débuté en 1998, avec des discussions entre les créateurs responsables de trois langages statistiques actuels (S, **R**, et Lisp-Stat). Le projet *Omegahat* développe également une collection de modules pour soutenir les nouvelles directions dans la programmation dans le langage S (comme mis en application dans **R** ou dans S-Plus). Les modules illustrent comment communiquer entre **R** et d'autres langages et applications. Il existe des modules pour traduire en **R** du code écrit dans un autre langage, ou pour faire appel à des fonctions **R** depuis un autre langage ou application. Ceci englobe aussi la possibilité d'inclure **R** dans un tel module, ou vice versa.

Conclusion

Dans ce document nous avons brièvement illustré les interfaces existantes de **R** avec des bases de données relationnelles comme étant un premier pas des logiciels statistiques modernes et libres, comme **R**, pour regagner le domaine du KDD. De plus, n'oublions pas que **R** fournit une grande variété de techniques statistiques et graphiques, et est fortement extensible. Le langage de S est souvent le véhicule de choix pour la recherche dans la méthodologie statistique, et **R** fournit un itinéraire libre pour participer à cette activité. Dans un milieu scientifique tel que l'EPFL, c'est à se demander si **R** ne devrait pas littéralement remplacer S-Plus, compte tenu des avantages de **R** et, en particulier, des interfaces qui lui sont possibles. Comme premier pas dans cette direction, nous espérons toujours que **R** sera prochainement disponible sur les serveurs ASIS, Distrilog et Cyclope du SIC.

Pour souligner ce propos, un deuxième article dans ce Flash informatique spécial montre l'utilisation de la géostatistique dans un monde libre en combinant **R** avec GRASS GIS.



Bibliographie

- [1] Diego Kuonen et Valerie Chavez-Demoulin (2001), **R** - un exemple du succès des modèles libres, Flash Informatique 2/01, <http://sic.epfl.ch/publications/FI01/fi-2-1/2-1-page3.html>
- [2] *The R Project for Statistical Computing*, <http://www.r-project.org>
- [3] *The Comprehensive R Archive Network (CRAN)*, <http://cran.r-project.org>
- [4] Michael J. A. Berry et Gordon Linoff (1997), *Data Mining Techniques: For Marketing, Sales, and Customer Support*, John Wiley & Sons.
- [5] Statoo.com - The Portal to Statistics on the Internet, <http://www.statoo.com>
- [6] Diego Kuonen's Data Mining Links, <http://bookmarks.kuonen.com/Statistics/Datamining/>
- [7] Daryl Pregibon (2001), *A statistical odyssey*, In Proceedings of KDD Conference '99.
- [8] Information, Support, and Software from the Hierarchical Data Format (HDF) Group of NCSA, <http://hdf.ncsa.uiuc.edu>
- [9] MySQL Homepage, <http://www.mysql.com>
- [10] David Axmark, Michael Widenius, Jeremy Cole, and Paul DuBois (2001), *MySQL Reference Manual*, <http://www.mysql.com/documentation/mysql>
- [11] Paul DuBois (2000), *MySQL. New Riders*
- [12] Hughes Technologies - The Home of Mini SQL (mSQL): <http://www.hughes.com.au>
- [13] PostgreSQL Homepage: <http://www.postgresql.org>
- [14] Interactive/On-line SQL Beginner and Advanced Tutorials: <http://sqlcourse.com> and <http://sqlcourse2.com>
- [15] **R Development Core Team** (2001), **R Data Import/Export**, <http://www.r-project.org>
- [16] The Omegahat Project for Statistical Computing, <http://www.omegahat.org>
- [17] Brian D. Ripley and Ruth M. Ripley (2001), *Applications of R clients and servers*, In Proceedings of the 2nd International Workshop on Distributed, Statistical Computing, March 15-17, Vienna, Austria, <http://www.ci.tuwien.ac.at/Conferences/DSC-2001>
- [18] Roger Koenker and Álvaro A. Novo (2000), *Statistical Analysis of Large Datasets - An Exploration of R - MySQL Interface*, <http://www.econ.uiuc.edu/~roger/research/rq/LM.html>
- [19] Brian D. Ripley (2001), *Using databases with R*, *R News*, 1(1): 18-20.
- [20] Torsten Hothorn, David A. James and Brian D. Ripley (2001), *R/S Interfaces to Databases*, In Proceedings of the 2nd International Workshop on Distributed, Statistical Computing, March 15-17, Vienna, Austria, <http://www.ci.tuwien.ac.at/Conferences/DSC-2001> ■

GRASS GIS ET R

MAIN DANS LA MAIN

DANS UN MONDE LIBRE



Reinhard.FURRER@epfl.ch & Diego.KUONEN@epfl.ch, EPFL, DÉPARTEMENT DE MATHÉMATIQUES



Rappelons que **R**, un exemple parmi tant d'autres du succès incontestable des modèles libres, est un langage et un environnement pour les calculs statistiques et leurs représentations graphiques. Cet article constitue le troisième paru dans le Flash informatique au sujet de **R**, cette année. Il conclut la trilogie sur **R**. Le premier article paru dans le FI 2/01 [1] était un point de départ pour **R**, tandis que le but du deuxième paru dans ce Flash informatique spécial [2] est d'illustrer les interfaces entre **R** et les bases de données relationnelles. Le but de ce troisième document est de montrer l'utilisation de la géostatistique dans un monde libre en combinant **R** avec GRASS GIS.

Ce texte n'est pas un manuel pour les systèmes d'information géographique, et nous ne prétendons pas être des spécialistes de ce domaine. C'est plutôt un regard de deux statisticiens pour souligner l'importance des logiciels libres dans notre domaine de recherche et de travail.

LES SYSTÈMES D'INFORMATION GÉOGRAPHIQUE

Les systèmes d'information géographique font partie de ces outils qui ont bénéficié des progrès réalisés dans le secteur de l'informatique. Le matériel et les logiciels sont de plus en plus performants et de plus en plus nombreux. Il est, aujourd'hui, difficile de concevoir que la réalisation d'un SIG (*Système d'Information Géographique*) se fasse sans les apports de l'informatique notamment pour la gestion, le traitement et l'analyse des données. Si les outils informatiques ont un rôle fondamental dans la conception des SIG, il faut toujours garder en mémoire qu'il ne s'agit que d'outils facilitant le travail du géographe. Bien connaître les logiciels de base de données géographiques ne suffit pas, il faut aussi et surtout connaître la démarche géographique et sa méthode.

Selon le *International GIS Dictionary* [3], un SIG ou un *GIS (Geographical Information System)* est un *computer system for capturing, managing, integrating, manipulating, analysing and displaying data which is spatially referenced to the Earth*.

C'est donc un ensemble organisé de matériel informatique, de logiciels, de données géographiques et de personnes capables de saisir, stocker, mettre à jour, manipuler, analyser et présenter toute forme d'information référencée géographiquement.

Ce qui distingue les SIG des autres formes de systèmes d'information, telles que les bases de données ou les tableurs, est qu'un SIG traite des informations spatiales. Un SIG possède la capacité d'associer plusieurs couches de données pour des points situés dans l'espace. L'information spatiale utilise une localisation avec un système de coordonnées ou système de référence (par exemple, un point est spécifié par la latitude et la longitude).

À l'origine, tout SIG repose sur une base de données géographiques. Un SIG n'est donc pas un simple système de cartographie assisté par ordinateur. S'il doit permettre la représentation des données qu'il gère et qu'il crée, celle-ci n'est pas impérativement cartographique. Les sorties peuvent être aussi alphanumériques et graphiques. Sans *Système de Gestion de Base de Données Localisées (SBGDL)*, pas de base de données géographiques. C'est le SGBD qui permet à l'utilisateur d'agir en entrée du système et en sortie.

Quelques applications des SIG sont évidentes, par exemple les administrations peuvent utiliser des SIG pour contrôler les bornes de propriétés ou estimer les ressources environnementales. Des SIG peuvent également être employés pour planifier des services tels que la santé et l'éducation primaire, en tenant compte de la distribution de la population et de l'accès aux équipements. Des SIG sont de plus en plus employés pour aider des entreprises à identifier leurs marchés potentiels et en mettant à jour une base de données spatiales de leurs clients.

De manière globale, un SIG peut être défini comme un ensemble de principes et techniques utilisés pour réaliser des objectifs comme:

- trouver des localisations ayant des attributs spécifiques. Par exemple trouver l'endroit pour un nouveau aéroport.
- examiner des attributs géographiques d'une localisation spécifique. Par exemple examiner la densité des routes d'une région particulière.

Les données des SIG sont souvent sauveées dans une ou plusieurs couches pour éviter des problèmes techniques causés par le traitement de grosses quantités de données. Il est plus facile de travailler avec des problèmes spatiaux complexes une couche à la fois pour permettre la révision des données sans devoir réviser le système d'information en entier.

Des données spatiales consistant en localisations et attributs sont représentées dans un SIG par les formats suivants:

- modèle vectoriel, sous forme d'objets géométriques: points, lignes, polygones qui représentent l'endroit et les bornes des entités géographiques;
- modèle *raster*, sous formes d'images composées de pixels.

Ces deux modèles ont des avantages et des désavantages. Le choix du modèle utilisé est souvent imposé par la nature du travail et les données disponibles.

Pour conclure cette section, voici quelques spécificités des données géographiques et spatiales:

- les observations ne sont pas indépendantes;
- les erreurs ont souvent une structure spatiale;
- les distributions sont non normales;
- la plupart des couches sont des données catégorielles ;
- les données sont rarement stationnaires;
- il y a souvent une interaction du temps avec l'espace, et
- il y a des données redondantes.

GRASS GIS

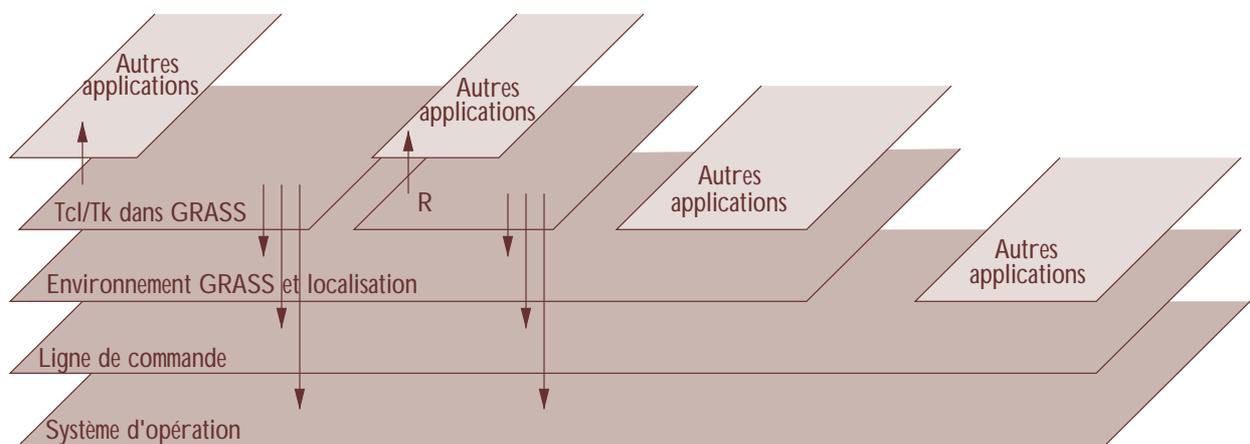


FIGURE 1 – ARCHITECTURE DE GRASS

GRASS (*Geographic Resources Analysis Support System*) est un système d'information géographique qui a été originellement programmé par des ingénieurs de l'armée américaine, plus précisément par l'*U.S. Army Construction Engineering Research Laboratories* (USA-CERL). Depuis, ce logiciel est maintenu et amélioré par une nouvelle armée de programmeurs, et maintenu officiellement par l'Université de Baylor aux États-Unis et l'Université d'Hanovre en Allemagne. Les utilisateurs de GRASS incluent la NASA, la NOAA, le USDA, le U.S. National Park Service, le U.S. Census Bureau, le USGS, et d'autres bureaux de *consulting environmental*; voir [4] pour plus de détails.

Les caractéristiques principales de GRASS sont:

- distribué librement sous licence GPL;
- disponible pour les principales architectures et systèmes d'exploitation (Unix/Linux, MacOS/X), avec une version préliminaire pour Windows NT/2000;
- lancé par une commande en ligne standardisée ou par une interface-utilisateurs graphique intuitive basée sur Tcl/Tk;
- accès à une DBMS (PostgreSQL, ou par ODBC, ...);
- 121 projections géographiques implantées; et support d'un vaste nombre de différents formats des données (*raster* et *vector*).

Comme GRASS est utilisé dans plusieurs universités, on a accès à plusieurs tutoriels et manuels [5, 6]. Comme mentionné, GRASS est lancé depuis une commande en ligne. Chaque commande GRASS n'est rien d'autre qu'un script *shell*. On peut donc lancer d'autres applications dans l'environnement GRASS, comme par exemple **R**. La figure 1 montre l'architecture de GRASS avec les différents niveaux des applications.

A titre d'exemple, lançons GRASS avec les données Spearfish qui est un jeu de données libre sur lequel la plupart des manuels se basent [7]:

```
$FreeWorld$ grass5
```

Le programme demande de spécifier la *location* et le *mapset*. Une *location* définit les bornes et les projections d'un projet. A chaque projet (i.e. location) on peut associer des *mapsets* définissant des sous-régions.

Ouvrons une fenêtre graphique pour afficher l'image topologique de la région définie par Spearfish et superposons la région du Mount Rushmore (ci-après Mount Redmont):

```
> d.mon start=x0
> d.rast map=elevation.dem
> d.rast -o map=redmont
```

À noter que GRASS ne met pas les fenêtres automatiquement à jour. Les fenêtres graphiques doivent être fermées avec `d.mon stop`. De plus, il n'est pas conseillé de changer leur taille interactivement avec la souris.

L'aide pour la commande `ps.map` (génération des fichiers postscript) est obtenue par:

```
> g.manual entries=ps.map
```

Les modules et commandes de GRASS sont structurés grâce à un préfixe; voir le tableau ci-après.

Préfixe	Classe	Signification des commandes
d.	display	présentation graphique et requêtes visuelles
s.	sites	traitement des données semis de points
r.	raster	traitement des données en mode raster
i.	imagery	traitement des données en mode image
v.	vector	traitement des données vectorielles
g.	general	commandes générales des opérations des fichiers
m.	misc	commandes diverses
p.	paint	commandes d'établissement de cartes
ps.	postscript	commandes d'établissement de cartes postscript

Presque toutes les commandes de GRASS peuvent aussi être effectuées par l'interface Tcl/Tk (lancée depuis GRASS avec la commande `tcltkgrass`). Par exemple pour rajouter les sites archéologiques à l'image topologique on peut, au lieu de taper la commande:

```
>d.sites type=box sitefile=archsites color=red size=6
```

utiliser le menu Site -> Display -> Display Site Markers qui ouvre la fenêtre montrée dans la Figure 2.



FIGURE 2 – SAISIE DE L'INTERFACE Tcl/Tk

UN PETIT EXEMPLE D'ANALYSE GÉOSTATISTIQUE

Une grande partie des travaux faits avec GRASS consistent en l'analyse et la manipulation des données, qui peuvent être faites avec un logiciel statistique comme **R**, conçu pour ce type de travail. La Figure 3 montre le domaine d'application de **R** dans GRASS.

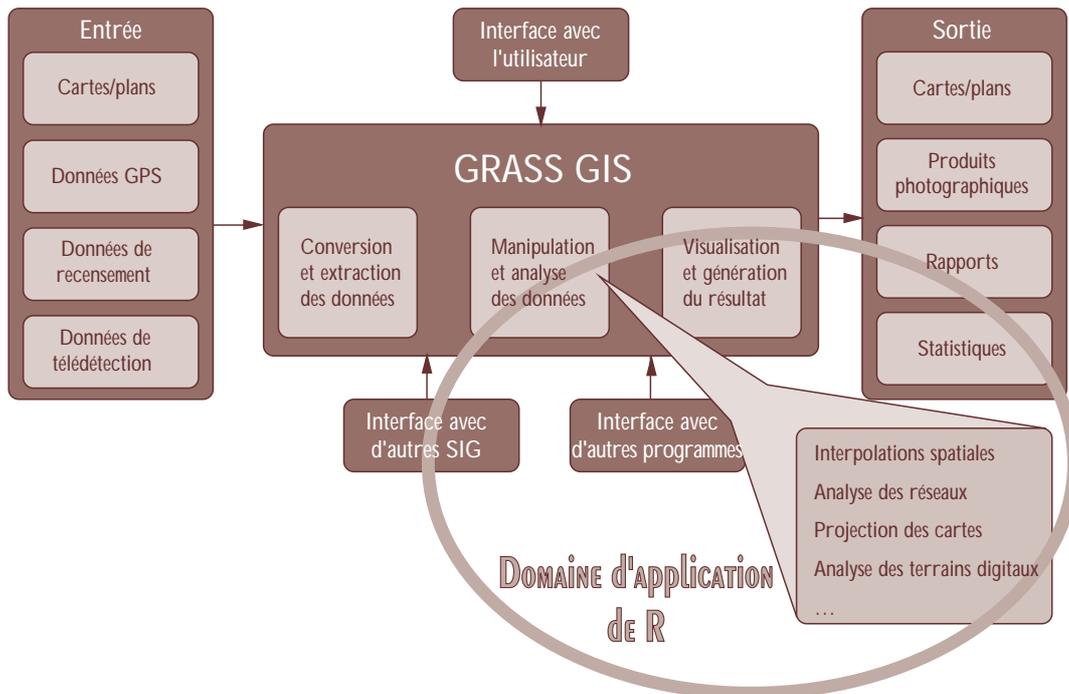


FIGURE 3 – DOMAINE d'application de **R** dans GRASS

Supposons qu'un travail consiste à comparer la densité des *bugs* (ci-après pingouins) avec une région comparable. La Figure 4 montre les pingouins dans une représentation tridimensionnelle (fait par la commande `nviz`). On peut voir les endroits comme des réalisations d'un processus ponctuel (inhomogène) - nous forçant de nous rappeler les statistiques spatiales. Pour des analyses géostatistiques comme l'interpolation spatiale, les corrélations spatiales, etc., il existe pour **R** plusieurs modules libres contenant ces fonctions [8].

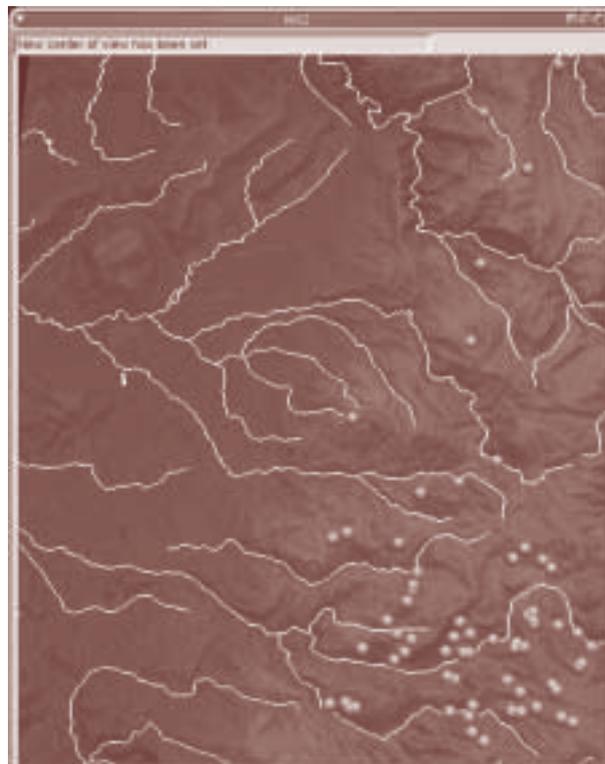


FIGURE 4 – LES ENDROITS DES PINGOUINS À LA CONQUÊTE DE MOUNT REDMONT

Pour l'utilisation de l'interface **R**/GRASS il faut installer le module GRASS, disponible à [9]. Il est recommandé d'installer aussi les modules *akima*, *VR*, *nlme*, *sgeostat*, *tripack* de **R** [10].

Reprenons GRASS avec les données Spearfish, lançons **R** et chargeons le module **R**-GRASS dans **R** (ayant lancé `tc1tkgrass` permet de travailler avec **R** et GRASS simultanément):

```
> R
> library(GRASS)
```

Dans la variable `G`, on affecte l'information géographique de notre jeu de données (pour les cracks, `G` est un objet de la classe `grassmeta`, ayant des méthodes `plot`, `summary`, etc.):

```
> G <- gmeta()
```

Nos données consistent en plusieurs différentes cartes *raster*, la commande suivante les affiche (pour ne pas alourdir l'article, on supprime la sortie de **R**):

```
> print(list.GRASS(type="rast"))
> print(list.GRASS(type="sites"))
```

Nous nous sommes intéressés aux pingouins et avons copié leurs positions dans la variable `world.domination` de **R**.

```
> world.domination <- sites.get(G, slist=c («bugsites»))[,1:2]
> names(world.domination) <- c('x', 'y')
```

Ripley [11] était le premier à réaliser l'importance de la fonction $K(r)$ (i.e. espérance du nombre évènements dans une disque de rayon r) pour décrire la dépendance spatiale. La fonction `khat` du module `splancs` [8] estime $K(r)$ dans un certain polygone.

```
> library(splancs)
> boundary <- cbind(x=c(590000, 609000, 609000, 590000), y=c(4914000, 4914000, 4928000, 4928000))
> distance <- seq(0, to=10000, by=200)
> k.hat <- khat(world.domination, boundary, distance)
> plot(distance, k.hat, main='La fonction K')
> lines(distance, pi*distance^2)
```

La Figure 5 montre le résultat. Malheureusement il n'est pas toujours facile d'interpréter cette fonction. Quelquefois, il suffit de construire une simple interpolation de la densité (Figure 6).

```
> plot(G, kde2d.G(world.domination$x, world.domination$y, h=c(2500, 2500), G=G), xlab='est',
                                             ylab='nord')
> title(«Densité des pingouins ('world.domination')»)
> points(world.domination)
> polygon(boundary)
```

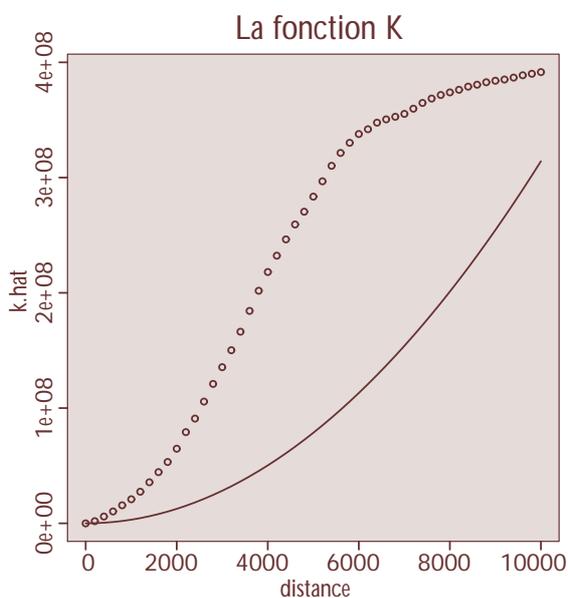


FIGURE 5 – LA FONCTION $K(r)$ ESTIMÉE

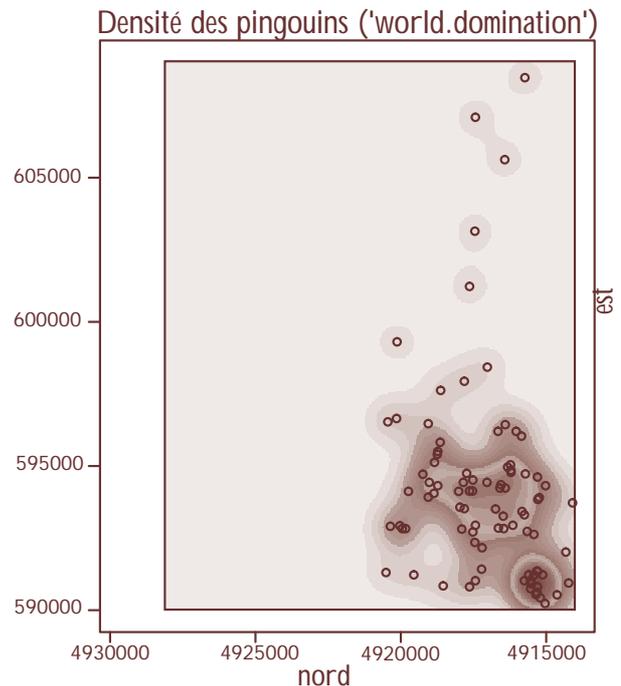


FIGURE 6 – LA DENSITÉ ESTIMÉE DES PINGOUINS À LA CONQUÊTE DU MOUNT REDMONT

POUR ALLER PLUS LOIN

L'architecture de GRASS (et de **R**) qui permet d'utiliser plusieurs programmes en parallèle (voir Figure 1), est illustré ici par quelques exemples particuliers.

Le modèle de mémoire statique de **R** peut causer des problèmes dans le cas de bases de données immenses. Pour y remédier, on peut utiliser le module `RPGSQL` de **R** [2], qui permet de lire et écrire des tables dans et depuis une base de données. GRASS

lui-même met aussi à disposition des interfaces à des DBMS ou des SGBD (*Data Base Management Systems* ou *Système de Gestion de Base de Données*) et à PostgreSQL; voir [12] pour plus de détails.

Un deuxième exemple est le programme `xfig` qui permet de créer des figures postscript. Pour rajouter à une carte de GRASS des flèches, du texte ou d'autres éléments graphiques, on ouvre `xfig` avec l'image transformée en format tiff ou exportée par la commande `v.out.xfig`.

A première vue, on pourrait argumenter sur le fait que GRASS n'est pas un programme ayant tous les éléments nécessaires pour un bon SIG: du point de vue des capacités statistiques et graphiques, de l'accès intégré aux bases de données, etc. Est-ce que GRASS est *sous-développé* ou incomplet? Non, au contraire, dans le concept et l'idéologie d'un monde libre chaque programme s'occupe de son point fort et au lieu d'inventer la roue deux fois, on laisse - grâce à une architecture modulaire - les tâches particulières aux programmes conçus pour ces problèmes. Ceci selon le proverbe: *Schuster bleib bei deinen Leisten* que l'on pourrait traduire par *À chacun son métier*.

Conclusion

Dans cet article nous avons brièvement illustré l'utilisation de la géostatistique en combinant **R** avec GRASS. Résumons quelques points faibles et forts de notre liste subjective.

Points faibles de l'utilisation de **R** et de GRASS:

- L'utilisation de GRASS n'est pas évidente même avec Tcl/Tk. Il faut avoir une certaine expérience avec des SIG.
- Il est difficile de créer des images postscript de manière efficace. Il faut souvent un post processing (par exemple avec `xfig`).
- La composante temps n'est pas encore prise en compte.

Par contre, on peut profiter des points forts suivants (on remarque que la plupart de ces points restent valables pour tous les logiciels libres:

- La conception du programme permet d'utiliser plusieurs programmes en parallèle.
- Il y a une abondante documentation et de nombreux exemples traités online (des cours universitaires y compris).
- Pour un statisticien les environ 250 commandes et les modules supplémentaires suffisent largement pour le travail. Sinon la conception de GRASS permet facilement d'élargir ou de rajouter des fonctions personnalisées.
- And last but not least: it's free ☺.

Bibliographie

- [1] **Diego Kuonen** & **Valerie Chavez-Demoulin** (2001), **R** - un exemple du succès des modèles libres, FI/02/01, <http://sic.epfl.ch/publications/FI01/fi-2-1/2-1-page3.html>
- [2] **Diego Kuonen** & **Reinhard Furrer** (2001), *Data mining avec R dans un monde libre*, FI-sp-01, <http://sic.epfl.ch/publications/FI01/fi-sp-01/>
- [3] **McDonnell Rachael** & **Kemp A. Karen** (1995). *International GIS, Dictionary*, Cambridge: GeoInformation International.
- [4] GRASS Homepage: <http://www.geog.uni-hannover.de/grass/> & Information générale de GRASS: <http://www.geog.uni-hannover.de/grass/general.html>
- [5] **Markus Neteler** (2001), *Short Introduction to Geostatistical and Spatial Data Analysis with GRASS and R statistical data language*, http://www.geog.uni-hannover.de/grass/statsgrass/grass_geostats.html
- [6] **Markus Neteler** (2000), *GRASS-Handbuch, Der praktische Leitfaden zum Geographischen Informationssystem GRASS*. Geosynthesis 11, Geographisches Institut der Universität Hannover (ISBN 3-927053-30-9).
- [7] Spearfish database description (postscript, 8 pages): <http://www.geog.uni-hannover.de/grass/gdp/tutorial/spearDB.ps.gz>
- [8] Module geoS: <http://www.maths.lancs.ac.uk/~ribeiro/geoS.html>, module spatstat: <http://www.maths.uwa.edu.au/~adrian/spatstat.html>, module splancs: <http://www.maths.lancs.ac.uk/~diggle/>, module spatial: <http://www.stats.ox.ac.uk/pub/MASS3/R.shtml>, module spat01: <http://www.cnr.colostate.edu/~robin/>
- [9] Module GRASS pour **R**: <ftp://reclus.nhh.no/pub/R/>
- [10] CRAN Package Sources: <http://cran.r-project.org/src/contrib/PACKAGES.html>
- [11] **Brian D. Ripley** (1976), *The second-order analysis of stationary point processes*, *Journal of Applied Probability*, 13, 255-266.
- [12] **Roger Bivand** & **Markus Neteler** (2000), *Open Source geocomputation: using the R data analysis language integrated with GRASS GIS and PostgreSQL data base systems*, <http://reclus.nhh.no/ge00/ge009.htm>
- [13] Free GIS Software and Free Geo-Data: <http://www.freegis.org>
- [14] SAL (Scientific Applications on Linux) - Geographic Information Systems: <http://sal.kachinatech.com/E/6/index.shtml>
- [15] Et si vous rêvez de télédétection plutôt que des SIG, alors voilà des logiciels libres pour vous: <http://www.remotesensing.org>, <http://www.ossim.org> ■

ENTRETIEN AVEC MÉLANIE CLÉMENT-FONTAINE

à propos de la traduction de la GNU GPL

Loïc DACHARY, ASSOCIATION FSF EUROPE – CHAPITRE FRANCE,
LoicDACHARY@fsfEUROPE.ORG



Mélanie Clément-Fontaine est doctorante en droit de la propriété intellectuelle et employée de la société Alcôve. Elle est l'auteur d'une étude juridique de la licence Publique Générale GNU – <http://crao.net/gpl/> – et participe à des conférences au sujet du droit et du Logiciel Libre (Colloque Institut National des Télécommunications – INT) Logiciels et Contenus Libres – http://www.int-evry.fr/colloque_libre/bio_mcf.html –, Libre Software Meeting – http://lsm2000.abul.org/program/topic17_fr.html –, Libre et vie locale – <http://libre.enst-bretagne.fr/LibreEtVieLocale/pub/programmeHoraire.html> –, ...).

Comment a été lancé le projet de produire une version française de la GNU GPL?

À la suite de mon intervention au Libre Software Meeting de Bordeaux en juillet 2000, j'ai eu une discussion avec Richard M. Stallman sur la nécessité d'adapter la licence GNU GPL au droit français. L'argumentation a continué ensuite par mail pour aboutir au début de l'année 2001 à une demande de sa part. Il souhaitait que je me charge de l'adaptation au droit français et j'ai accepté.

Comment concrètement avez-vous procédé?

J'ai tout d'abord demandé à Frédéric Couchet s'il m'était possible d'obtenir de l'aide sur la traduction technique. Il m'a dirigé vers Benjamin Drieu. Nous avons commencé à travailler ensemble sur la base des traductions existantes.

La complexité du travail m'a surprise. Et ce qui me gêne le plus est d'être la seule juriste française à réfléchir sur le sujet. Mon effort actuel représente en tout quelques jours de travail.

Il n'est pas possible de se contenter d'une traduction mot à mot et Richard M. Stallman a été explicite sur ce point. Il y a des passages dont la signification me gênait et si on les changeait cela était assez important pour que nous ne puissions pas faire la modification seuls. Nous avons préparé des corrections que nous devons soumettre à Richard M. Stallman et Eben Moglen. Il est aussi nécessaire que nous ayons une visibilité sur la version 3 de la GNU GPL afin d'éviter de perdre tout ou partie de notre travail d'adaptation.

Où en êtes-vous du processus de traduction?

Nous avons actuellement une version correcte en langage commun, compilée à partir des traductions existantes. Cette étape est accomplie. Nous avons aussi identifié les formules ayant un caractère juridique et il faut travailler sur chacune d'entre elles pour y apporter les corrections propres au langage juridique.

J'ai conservé tous nos documents de travail dont certains sont sur papier.

La GNU GPL s'est fondée sur la convention de Berne?

Oui, mais elle ne lie que les états et non les individus. Elle ne peut pas être invoquée par les individus. Elle établit quelques règles qui organisent la réciprocité: une œuvre protégée par un état sera aussi protégée par la législation des autres états. Ensuite l'état édicte les règles qui lient les citoyens. Si l'état n'édicte pas ces règles, il est coupable vis à vis des autres états. Les États-Unis et la France sont signataires. Mais chaque état peut établir des réserves. Par exemple les États-Unis ont émis une réserve sur l'article 6 bis qui touche au droit moral. L'étape de transcription sous forme de règles propres à chaque état laisse place à des interprétations qui rendent l'homogénéité de l'ensemble assez aléatoire. La convention de Berne (1886 et modifiée de nombreuses fois) est très courte et le droit français y correspond. Je ne me suis pas posé de questions relatives à la convention de Berne lors de mon travail préliminaire à la traduction de la GNU GPL. Il s'agit de l'adapter au droit français et non à la convention de Berne à laquelle elle est déjà fidèle.

Comment la version française s'appliquera dans les pays de langue étrangère? En Allemagne par exemple?

C'est délicat. En Europe cela sera facilité grâce à l'harmonisation à venir. C'est pour cela qu'il est plus simple de dire dans la licence quelle est la juridiction applicable. Sinon il sera toujours problématique de déterminer quelle est la juridiction et la loi compétentes. Qui est le consommateur? Il y a des textes qui protègent le consommateur et qui sont souvent appliqués, il faut les prendre en compte. Richard M. Stallman n'est pas favorable à l'idée de préciser la loi et la juridiction compétentes dans la GNU GPL.

La GNU GPL en français sera compatible avec la GNU GPL en anglais?

Bien sûr, c'est l'objet de ce travail.

Comment pensez-vous que la FSF Europe peut aider?

Nous avons besoin d'être en contact avec d'autres juristes. Pour les relectures, l'étude de chaque point de vocabulaire. La FSF Europe peut nous aider à organiser les réunions de travail et nous mettre en contact.

Je pense que l'organisation du travail serait efficace sous la forme de réunions de travail regroupant les personnes ayant des compétences juridiques pour tenter d'éclaircir un point précis. Je peux proposer le premier ordre du jour et ensuite chaque réunion déterminera l'ordre du jour de la réunion suivante.

■

Copyright © 2001 FSF France, 8 rue de Valois, 75001 Paris, France [la reproduction exacte et la distribution intégrale de cet article sont permises sur n'importe quel support d'archivage, pourvu que cette notice soit préservée].



UNE PLATE-FORME D'INGÉNIERIE LOGICIELLE EN OPEN SOURCE



*M*atra Datavision, ou EADS Matra Datavision, sa nouvelle dénomination, fut leader sur le marché de la CFAO, des années quatre-vingts jusqu'au milieu des années quatre-vingt-dix grâce à ses activités de développement logiciel en France.

A partir de ses origines dans le développement aéronautique, EUCLID a réussi à prendre une part de marché considérable dans le secteur des applications mécaniques au sens large, de l'ingénierie mécanique générale aux industries complexes de l'automobile et de l'aéronautique. Pendant de nombreuses années, le produit était en concurrence directe avec CATIA, de temps en temps le surpassant largement en termes de certaines fonctionnalités. Bien avant la plupart de leurs concurrents, les ingénieurs du développement logiciel de Matra utilisaient le système de modélisation intégrant une connexion directe des données CAO avec les programmes de commande numérique. Les spécialistes basés sur Paris ont créé une plate-forme de développement appelée CASCADE pour leur permettre de créer une nouvelle génération de produits. Mais le projet EUCLID Quantum n'a jamais vraiment pris de l'essor. Ce leader dans l'édition de logiciels standards a dû admettre que les exigences d'un tel produit et le nombre d'ingénieurs de développement nécessaires grandissaient constamment tandis que les marges potentielles et les marchés diminuaient.

Ainsi, la décision fut prise en 1998 d'interrompre le développement de logiciels standards. Une partie du système et un certain nombre d'employés se dirigea vers Dassault Systèmes, et le groupe Matra Datavision lui-même devint le partenaire commercial clé d'IBM pour le marketing de CATIA. Le passage au nouveau *business model* qui met l'accent sur le service et le *consulting* s'est révélé être un succès l'année suivante. Aujourd'hui Matra est le partenaire commercial privilégié d'IBM pour les solutions de gestion de cycle de vie produit, sur des pro-

jets liés aux logiciels GDT, EUCLID, CATIA, ENOVIA et DENEb. Et la société est aujourd'hui leader mondial dans les ventes de la nouvelle Version 5 de CATIA. Jusqu'ici rien de particulier. Un certain nombre d'autres pionniers dans le secteur du développement logiciel ont interrompu leur développement interne et orientent aujourd'hui leur stratégie commerciale autour de produits externes. Cependant, la décision prise par Matra un an plus tard et annoncée en décembre 1999 lors d'une conférence de presse internationale à Paris annonçait une nouveauté: l'environnement de développement EUCLID Quantum serait disponible gratuitement sur Internet sous le nom d'Open CASCADE.

UN SERVICE BASÉ SUR LOGICIEL LIBRE PLUTÔT QUE SUR UN PRODUIT

Les participants à la conférence ont été quelque peu étonnés et fortement sceptiques au départ. Il y avait, bien sûr, l'exemple de Linux, une version libre de système d'exploitation, devenu un sérieux concurrent des produits Microsoft Windows, ceci grâce à la coopération de la communauté de développeurs internationale sur Internet depuis de nombreuses années. Mais Linux, à l'instar d'autres exemples Open Source, vise un public plus large et par conséquent un plus grand nombre de développeurs. En comparaison, le secteur de la CFAO est petit et l'idée d'y utiliser l'Open Source était complètement méconnue à cette époque.

En outre, notamment en Europe, les utilisateurs d'une discipline technique sont davantage impressionnés par un logiciel standard de qualité offrant une garantie de fiabilité. De plus, les systèmes du marché ont atteint un tel niveau de maturité qu'ils servent pratiquement tous les secteurs d'application. Et rien ne laissait présager que de nouvelles sociétés

seraient créées dans le domaine du logiciel CAO. Qui alors, hormis des instituts de recherches et des universités, serait intéressé par le téléchargement via Internet d'une vaste bibliothèque de composants et des outils de base? Et à quoi servira un développement avec Open CASCADE?

En premier lieu, regardons ce qui a motivé Matra à prendre cette décision. CASCADE avait déjà mené une double existence pendant de nombreuses années. D'une part, c'était le cœur du logiciel EUCLID et les briques pour le bâtir; d'autre part, Matra l'utilisait pour une série de projets d'applications spécifiques.

Parfois Matra était maître d'œuvre, parfois les clients achetaient eux-mêmes CASCADE pour réaliser leurs propres applications.

Dans chaque cas, le projet ne pouvait être achevé avec un logiciel standard existant. Certains des projets concernaient des applications mécaniques, mais ceux-ci ne représentaient qu'une petite part, plus ou moins insignifiante, de la totalité. Ces projets exigeaient typiquement un lien entre des données géométriques extrêmement diverses, généralement de la modélisation 3D, à d'autres informations.

A titre d'exemple, une compagnie pétrolière voulait créer un modèle 3D du sous-sol géologique à partir d'échantillons représentatifs, ce qui aurait permis de déterminer les options de traitement ainsi que les contraintes. Autre exemple: une usine électrique désirait une représentation réaliste et mesurable d'un réseau de lignes à haute tension pour visualiser l'impact des changements climatiques et des variations thermiques.

Dans le domaine de l'ingénierie médicale, les prothèses seront de plus en plus individualisées à l'avenir. Cependant, les contours des parties du corps doivent être parcourus et calculés pour que, par exemple, des articulations artificielles puissent être automatiquement réalisées par des programmes NC à 5 axes. Le logiciel standard n'a pas été conçu pour répondre à de telles exigences et une programmation complémentaire est souvent bien trop onéreuse. De nouveaux éléments doivent souvent être définis, exigeant une intervention dans la structure de données. Avec CASCADE, ces problèmes pourraient être résolus avec succès. Toutefois, vendre

CASCADE pour ce type de développement était non seulement insignifiant dans le cadre global d'un projet, mais représentait plutôt une entrave aux négociations contractuelles avec les parties intéressées. Après tout, pourquoi *pinait* sur 50 000 marks allemands quand des millions sont en jeu?

La publication et la libre distribution de CASCADE apporterait beaucoup de bénéfices à ce type de projet. En particulier, grâce à un accès mondial, le cercle de prospects et de clients potentiels pourrait s'accroître de manière exponentielle.

Le libre téléchargement d'Open CASCADE pouvait signifier que les spécialistes de développement de Matra deviennent superflus. Mais ce souci n'est pas fondé. Des années d'expérience ont démontré que si le savoir-faire métier est très répandu

parmi les utilisateurs, ce n'est généralement pas le cas pour la compétence à assurer une programmation optimale.

Une seconde motivation repose sur les possibilités de développement offertes par Open CASCADE. En raison de la disponibilité gratuite de la plate-forme pour créer des applications spécifiques et des possibilités de modification et d'extension de celle-ci en y ajoutant de nouveaux composants et de nouvelles fonctions, un grand nombre de spécialistes logiciels viendrait élargir l'équipe restreinte de développeurs internes. En raison de son savoir-faire avancé, Matra a ainsi été prédestiné pour le rôle d'un *Red Hat Open CASCADE*, susceptible de fournir des versions certifiées à intervalles spécifiques à une communauté internationale de développeurs. Début 2000, Matra a mis en place un comité de pilotage qui concentre ses efforts sur l'intégration de compétences internationales. Cette équipe comptait 23 participants de tous les continents avant la fin de l'année.

L'essor inattendu d'un nouveau marché

En résumé: la décision courageuse a porté ses fruits. Le succès réalisé en une année a abouti à la fondation de la Société Open CASCADE en décembre 2000.

Pendant son premier mois d'existence, le site ***opencascade.org*** a enregistré plus de 1'500



téléchargements. Cela s'est passé en janvier de l'année dernière. Avant décembre ce nombre augmenta considérablement pour atteindre 11'000 téléchargements. Une version complète représente environ 250 Mo. On peut donc supposer que ces téléchargements furent l'œuvre de personnes vraiment désireuses de réaliser un projet.

Environ 30'000 visiteurs se rendent chaque mois sur le site Internet, 300 fichiers Open CASCADE sont quotidiennement téléchargés et ces deux tendances continuent d'augmenter.

Suite à la sortie de la première Version 3.0 certifiée en avril, plus de 5'000 fichiers ont été téléchargés en une semaine. Et la réaction de la communauté fut encore plus enthousiaste en novembre avec la sortie de la Version 3.1. La liste des pays où Open CASCADE intervient dans la réalisation de projets regroupe plus de 60 nations dans le monde entier. Les États-Unis sont en tête de liste avec environ 2'500 projets, suivis par la France avec 1'100, l'Allemagne avec environ 700 et l'Italie avec 500.

Les utilisateurs et les développeurs proviennent de trois secteurs: l'industrie générale, la R&D et l'industrie du logiciel. La version actuelle du logiciel inclut environ 2'000 classes de base, 20'000 fonctions et, bien sûr, la documentation pour l'utilisation professionnelle. Elle contient des composants C++ pour la structuration de modèles de données, Visual C++ pour la compilation du code Open CASCADE pour la plate-forme Windows, les algorithmes de modélisation, les outils de visualisation et d'échange de données et un ensemble d'outils de développement.

Outre STEP et IGES, les formats de sortie incluent désormais STL pour le prototypage rapide, aussi bien que VRML et HPGL2. Open CASCADE est le premier logiciel de CAO utilisé dans l'industrie sous Linux et il fonctionne également sur les plates-formes Windows, Sun Solaris, SGI et IBM/AIX. La nouvelle société Open CASCADE, filiale d'EADS Matra Datavision, regroupait plus de cent personnes vers la fin de l'année 2000, dont 80 ingénieurs logiciel basés sur les sites français et russes. Ce chiffre devrait augmenter pour atteindre 170 à la fin de 2001.

Pour étoffer l'assistance au développement et à la formation, la société Open CASCADE offre actuellement une large gamme de services qui inclut un helpdesk avec un suivi personnalisé, des mises à jour mensuelles, des interventions ponctuelles et le *consulting*.

La société a enregistré un chiffre d'affaires d'environ huit millions de marks allemands lors de son premier exercice. Et la direction prévoit une forte croissance dans les années à venir.

Au cours de cette même année où de nombreuses start-up furent surnommées *startdown*, leurs dirigeants peu soucieux des lendemains, jetant des millions par la fenêtre, la Société Open CASCADE a fourni un merveilleux exemple de réussite sur la base de l'Internet et du logiciel libre. Son objectif principal est de fournir technologie et services.

LA STANDARDISATION DES LOGICIELS ÉVOLUE DANS UNE NOUVELLE DIRECTION

Le succès repose sur deux facteurs: la standardisation logicielle et la demande croissante de l'industrie, des instituts de recherche et des vendeurs de logiciel pour différencier leur offre en utilisant des applications innovantes et fortement spécialisées. Ce qui est nouveau c'est que la technologie sous-jacente, pour laquelle Open CASCADE peut devenir la norme, couvre un secteur fortement spécialisé et fortement complexe, à savoir la création et le traitement d'information graphique 3D. A la différence du *Knowledge ware* de Dassault Systèmes, qui offre la possibilité de générer une valeur ajoutée uniquement à CATIA V4, Open CASCADE est indépendante du système de l'utilisateur.

BMW, par exemple, est l'un des plus grands comptes d'IBM, cependant la société a souhaité l'année dernière adopter le nouveau logiciel Open Source de Matra pour développer une application appelée Quick Mesh. Ce projet comprend la génération de maillages à partir de données de CATIA V4 et d'autres modèles bien adaptés aux études aérodynamiques. Grâce au nouvel environnement de développement, il était possible d'adapter ce générateur de maillage aux exigences spécifiques de BMW pour des analyses de flux. Dassault Systèmes a lui-même utilisé l'ensemble des outils de Matra pour développer une interface directe entre EUCLID 3 et CATIA V4.

Un autre projet a aussi montré que les méthodes de développement de logiciel évoluent grâce à la disponibilité de l'Open Source sur Internet.

Ce projet s'appelle SALOME. Les participants représentent actuellement des entreprises industrielles, des instituts de recherches et des éditeurs de logiciel français. Le projet a été certifié par le RNTL, le Réseau National de Technologie de Logiciel et

financé par le ministère de l'Industrie. La Société Open CASCADE dirige ce projet.

De nouveau, les participants aspirent à réaliser une structure logicielle ouverte permettant la liaison CAO à l'analyse et aux applications de simulation, ainsi que l'intégration de solveurs existants. Ce projet se déroulera en trois phases et se terminera fin 2002.

Cette solution logicielle ouverte basée sur un réseau ouvert promet de raccourcir et de faciliter considérablement le processus de développement. Avec la mise en commun des compétences de programmation, cette solution apporte les mêmes bénéfices à tous les participants. L'ancien éditeur de logiciels standards se concentre désormais sur la standardisation d'outils de développements logiciels et prévoit un support technique aux industries partici-

pantes. C'est nouveau, mais cela ne restera pas un phénomène isolé. La technologie de l'information a atteint un niveau où la disponibilité logicielle sur Internet devient chose courante.

Cependant, les bénéfices apportés pour les tâches d'applications spécifiques seront clés. La société qui peut offrir les technologies, les outils adéquats, et bien sûr, le savoir-faire et le conseil jouera un rôle majeur dans les décennies à venir. La Société Open CASCADE fera sûrement partie de celles-ci.

Open CASCADE – An EADS Matra Datavision Company

© Ulrich Sendler, Springer-Verlag, 2001

Ndlr: Madame Isabelle Maillot,
i-maillot@paris1.matra-dtr.fr
est disposée à venir présenter ce logiciel
aux personnes intéressées. ■



COPYRIGHT ET MONDIALISATION À L'ÂGE DES RÉSEAUX INFORMATIQUES



RICHARD M. STALLMAN, RMS@GNU.ORG

TRADUCTEUR: SÉBASTIEN BLONDEEL, BLONDEEL@APRIL.ORG



NOTES DU TRADUCTEUR

Ce texte est la traduction en français de l'intervention de Richard Stallman [1], fondateur du projet GNU [2], le 19 avril 2001, dans le cycle de conférences *MIT Communications Forum* [3] organisées par le MIT.

Cette conférence se trouve sur le web à deux adresses de référence, sur le site du MIT [4] et sur le site web du projet GNU [5]. Le présent extrait se limite au discours magistral de Richard Stallman. Vous trouverez la traduction complète, incluant la séance de questions du public, pointée depuis les adresses [4] et [5].

GLOSSAIRE

L'orateur s'adressant à des États-Uniens comme lui, il fait de nombreuses références évasives à des sujets qui peuvent surprendre un lecteur européen. Nous tentons ici d'expliquer ces derniers en quelques mots, les moteurs de recherche sur le web étant là pour vous aider à approfondir le sujet si tel est votre désir.

copyright

C'est une notion de droit anglo-saxon qui n'a pas d'équivalent exact en droit latin, mais qui peut être approchée par le droit d'auteur. Nous avons donc gardé le mot anglais, qui est bien compris dans un texte écrit en français.

droit de citation

Dans la présente traduction, cette expression traduit incorrectement la notion de droit anglo-saxon de *fair use*. Il s'agit de l'usage d'un texte soumis à copyright ne nécessitant pas l'accord explicite du détenteur du copyright (par exemple, le droit de publier de courtes citations d'une œuvre ou celui de faire une copie privée sont garantis en France par le **Code de la propriété intellectuelle**, et le *fair use* est en quelque sorte l'analogie anglo-saxon de ces garanties).

DMCA

Digital Millennium Copyright Act peut se traduire par *loi du millénaire sur le copyright en matière d'œuvres numériques*. Loi votée par le Congrès des États Unis d'Amérique le 20 octobre 1998, qui introduit un durcissement notable de la législation sur le copyright, dans le but prétendu de lutter contre le *piratage* (c'est un mot que Richard Stallman rejette et déconseille d'utiliser) entraîné par les technologies numériques; notamment, toute tentative pour contourner un moyen de protection mis en place par le détenteur du copyright (comme le fait d'annuler sur un lecteur de DVD la protection géographique ou *zonage*) serait illégale selon cette nouvelle loi. C'est à cause de cette loi que le programmeur russe Dmitry Sklyarov a été arrêté à Las Vegas le 16 juillet 2001 alors qu'il assistait à une conférence internationale d'informatique, ce qui a provoqué de nombreuses réactions dans le monde. Vous trouverez plus d'informations sur ce sujet récent, ainsi que des liens vers divers sites de protestation ou ayant trait aux manifestations suite à cette arrestation, sur le site web de l'EFF [6], et vous lirez notamment avec intérêt la FAQ de cette affaire [7].

ALÉNA

Accord de Libre-Échange Nord-Américain est la traduction de NAFTA, *North American Free Trade Agreement*. Accord de libre-échange économique entre le Canada, les États-Unis d'Amérique et le Mexique. Les États-Unis d'Amérique sont critiqués car on estime qu'ils souhaitent en retirer tous les bénéfices sans en payer le prix.

Copie exacte

On a appelé *copie exacte* ce que l'anglais désignait par le mot *verbatim*, et on fera remarquer qu'il y parfois confusion entre les notions d'activité *commerciale* et à *but lucratif*— tout est *commercial*, même l'éducation, la distinction pertinente se fait plutôt sur le côté lucratif ou non de l'activité. Cette confusion a été *respectée* dans la traduction.

majors

majors, sans bon équivalent français, est repris tel quel; il représente les grandes multinationales de production des industries du cinéma et du disque.

Napster

Nom d'un site web qui centralisait et organisait les échanges de fichiers entre ordinateurs sur l'Internet par un système de *pair à pair*, avant que la société dont il dépendait ne soit attaquée en justice par les *majors* et qu'elle doive considérablement limiter et modifier ses activités. Vous trouverez plus d'informations sur l'histoire de cette affaire sur le site de l'EFF par exemple [8]. Bien entendu de par leur nature même les systèmes de *pair à pair* sont difficiles à fermer ou à contrôler; on peut citer dans une veine similaire, le projet Freenet [9].

REMERCIEMENTS

Le traducteur tient à remercier les personnes suivantes:

David Madore

... pour avoir défini à ma demande et en quelques lignes les termes du glossaire, définitions que je me suis borné à largement recopier.

Anne Possoz

... pour avoir contacté les orateurs pour leur proposer l'inclusion d'une traduction en français de ce texte dans une publication de l'EPFL, et pour avoir travaillé de concert avec moi sur cette traduction, notamment sur la relecture.

Richard Stallman

... pour tout ce qu'il a fait, et pour l'énergie qu'il continue à brûler sans compter sur les chevaux de bataille qui lui tiennent à cœur. On aurait tort de croire que l'introduction dithyrambique de l'animateur est excessivement flatteuse ou complaisante – et j'invite le lecteur qui ne les connaîtrait pas encore à se pencher sur l'œuvre de Stallman, accessible depuis son site personnel [1]

David Thorburn

... pour avoir rapidement accepté l'idée de traduction et proposé de mentionner cette version française depuis la zone web réservée à ce cycle de conférences. Nous devons également lui savoir gré d'avoir invité Richard Stallman et ce faisant, d'avoir apporté toute cette matière à réflexion.



TRANSCRIPTION

jeudi 19 avril 2001, 17:00 – 19:00



David Thorburn, animateur:

Notre intervenant d'aujourd'hui, Richard Stallman, est une figure légendaire dans le monde de l'informatique. Ce fut pour moi une expérience instructive de tenter de trouver un interlocuteur qui lui donne la réplique. Un professeur éminent du MIT m'a confié qu'il faut considérer Stallman comme une figure charismatique de parabole biblique – *«Imaginez», m'a-t-il dit, «un Moïse ou un Jérémie. Un Jérémie, plutôt.»* Ce à quoi j'ai rétorqué: *«Et bien, voilà qui est admirable. Voilà qui est merveilleux, et qui confirme mon impression de l'importance de sa contribution au monde. Pourquoi, dans ce cas, avoir une telle réticence à partager la tribune avec lui?»* Réponse: *«Tel Jérémie ou Moïse, il me submergerait, tout simplement. Je refuse de débattre avec lui, mais s'il me fallait nommer cinq personnes vivantes dont les actes ont véritablement bénéficié à l'humanité, Richard Stallman serait l'une d'elles.»*



Richard Stallman:

Il me faut commencer par expliquer pourquoi j'ai refusé d'autoriser que ce débat soit diffusé en direct sur le Web, au cas où le problème ne serait pas apparu clairement: le logiciel utilisé pour la diffusion sur le Web oblige l'utilisateur à télécharger un autre logiciel pour recevoir l'émission. Ce logiciel n'est pas libre. Il est disponible gratuitement mais uniquement sous forme exécutable, c'est-à-dire sous la forme d'une suite de nombres cabalistique. Ce qu'il fait est secret. On ne peut ni l'étudier, ni le modifier, et certainement pas en publier de

version modifiée. Et ce sont là des libertés essentielles dans la définition du logiciel libre.

C'est pourquoi, si je souhaite me comporter en prosélyte honnête du logiciel libre, je ne peux pas d'un côté donner des conférences, et de l'autre inciter les gens à utiliser des logiciels qui ne sont pas libres. Cela saperait ma propre cause. Si je ne prends pas mes principes au sérieux, comment m'attendre à ce que d'autres le fassent?

Cependant, je ne suis pas venu parler de logiciel libre. Après avoir travaillé plusieurs années pour le mouvement du logiciel libre et avec les premiers utilisateurs du système d'exploitation GNU, j'ai commencé à recevoir des invitations pour donner des discours, discours où on me posait des questions telles que: *«Les idées de liberté pour les utilisateurs de logiciels s'étendent-elles à d'autres domaines et comment?»*

Et bien sûr, on me posait de sottises questions comme *«Le matériel informatique devrait-il être libre?»* *«Ce microphone devrait-il être libre?»*

Qu'est-ce que cela peut bien vouloir dire? Devrait-on avoir la liberté de le copier et de le modifier? En ce qui concerne les modifications, personne ne peut s'opposer à ce que vous modifiez un microphone que vous auriez acheté. Quant à la copie, personne ne dispose d'un duplicateur de microphone – de telles choses n'existent que dans les histoires de science-fiction. Un jour, nous disposerons peut-être de nanotechniques d'analyse et d'assemblage, et il sera peut-être possible de copier pour de vrai un objet physique. Alors, les libertés attachées aux objets prendront une véritable importance. Nous verrons des fabricants de produits agricoles tenter d'interdire de dupliquer la nourriture, et il s'agira d'un débat politique de première importance – si de telles nanotechniques voient jamais le jour. J'ignore si ce sera le cas: il ne s'agit pour le moment que de pure spéculation.

Mais cette question se pose pour d'autres types d'informations: toute information stockable sur ordinateur

peut être copiée et modifiée. Par conséquent, les problématiques éthiques du logiciel libre, les droits qu'a l'utilisateur de copier et de modifier le logiciel, sont transposables à d'autres types d'informations. Soyons clairs: je ne parle pas des informations privées, telles que les informations personnelles, qu'il n'est jamais question de rendre publiques. Je traite des droits qu'il faudrait donner à quiconque reçoit des copies de données publiées, sans intention ni tentative de les garder secrètes.

Pour exposer mes idées sur la question, je vais passer en revue l'histoire de la distribution de l'information et du copyright. Jadis, les livres étaient écrits à la main à l'aide d'une plume, et tout lettré pouvait recopier un livre presque aussi efficacement que quiconque. Bien sûr, celui qui exerçait cette activité toute la journée y acquérait probablement des compétences particulières, mais la différence n'était pas énorme. Les copies étant exécutées à la pièce, l'industrialisation du procédé n'apportait rien. La réalisation de dix copies exigeait dix fois plus d'efforts que la réalisation d'une seule copie. Rien non plus n'incitait à la centralisation: on pouvait copier un livre n'importe où.

Cette technique n'impliquant pas la similitude des copies, on ne distinguait pas avec une telle insistance que de nos jours, les processus de création originale et de copie. Il existait des intermédiaires parfaitement raisonnables. On savait ce qu'était un auteur, on savait, par exemple, que telle pièce avait été écrite par Sophocle, mais on pouvait imaginer, entre la rédaction et la copie, d'autres opérations utiles. On pouvait par exemple recopier une partie d'un livre, ajouter de nouveaux passages, recopier une autre partie, ajouter d'autres passages, etc. On appelait cela *écrire un commentaire* – c'était fréquent – et ces commentaires étaient prisés.

On pouvait aussi copier un passage d'un livre, écrire d'autres mots, copier un passage d'un autre livre, écrire d'autres mots, etc. – on appelait cela écrire un recueil. Les recueils étaient fort utiles également. Certaines œuvres sont perdues et ne survivent que par le truchement de fragments repris dans d'autres livres qui ont eu plus de succès que l'original.

Peut-être seules les parties les plus intéressantes étaient-elles reprises, et ces sélections étaient abondamment recopiées, alors que personne ne prenait la peine de recopier les textes originaux, jugés trop peu intéressants.

Autant que je sache, rien ne ressemblait alors au copyright. Quiconque souhaitait copier un livre le pouvait. Plus tard, avec l'invention de l'imprimerie, des livres ont commencé à être imprimés. L'imprimerie ne représentait pas uniquement un progrès quantitatif dans la facilité de copier. Elle affectait de façon différente divers types de copies en introduisant une économie d'échelle intrinsèque. C'était un travail long et fastidieux que de préparer les planches de caractères amovibles, comparée à quoi l'opération d'imprimer la même page plusieurs fois était négligeable. Il en résulta une centralisation et une production en série des livres; ce n'est vraisemblablement qu'en quelques endroits qu'un livre donné pouvait être copié.

Il en résulta également que les lecteurs ordinaires ne pouvaient plus copier efficacement des livres. Cette activité était réservée à ceux qui disposaient d'une presse à imprimer: elle était devenue industrielle.

Cette révolution s'est opérée graduellement, et l'imprimerie n'a pas totalement remplacé la copie manuelle lors des premiers siècles de sa mise en service. Des riches comme des pauvres continuaient de recopier ou de faire recopier des livres à la main. Les riches, pour exhiber leur richesse par le truchement d'exemplaires richement enluminés; les pauvres, par manque d'argent pour acheter un livre imprimé et parce qu'ils avaient le temps. Comme on dit, *«le temps n'est pas de l'argent tant que l'on a du temps à revendre»*.

Les copies manuelles ont donc survécu quelque temps. Je pense que c'est au XIX^e siècle que les tarifs de l'imprimerie devinrent suffisamment accessibles pour que même les pauvres instruits puissent s'offrir des livres imprimés.

Le copyright fut développé avec l'essor de l'imprimerie, qu'il réglementa au sens industriel du terme. Il ne limitait pas les droits des lecteurs; il limitait les pouvoirs des éditeurs et des auteurs. En Angleterre, le copy-

right fut initialement une forme de censure. Il fallait obtenir la permission du gouvernement pour publier un livre. Mais l'idée évolua. Au moment de la constitution des États-Unis d'Amérique, on avait une autre idée du but du copyright, et je pense que cette idée fut reprise en Angleterre également.

Pour la constitution des États-Unis d'Amérique, on proposa que les auteurs obtinssent un copyright, c'est-à-dire un monopole sur la copie de leurs livres. Cette proposition fut rejetée, et remplacée par une proposition cruciallement différente: dans le but de promouvoir le progrès, le Congrès pourrait, s'il le souhaitait, établir un système de copyright qui créerait ces monopoles. Si l'on en croit la constitution des États-Unis d'Amérique, les monopoles n'existent pas pour le bien de ceux qui les détiennent; ils n'existent que dans le but de promouvoir le progrès scientifique. On attribue un monopole aux auteurs afin de les inciter à servir la population par leur œuvre.

Le but recherché était donc une augmentation du nombre de livres écrits et publiés, mis à la disposition du public. On pensait que cela inciterait les productions littéraires, scientifiques et d'autres domaines encore, et par voie de conséquence, l'instruction et l'éducation de la société. Voilà le but à servir. La création de monopoles privés n'était qu'un moyen pour servir une fin, et cette fin était le bien public.

À l'âge de l'imprimerie, le copyright était peu douloureux, puisque c'était un règlement industriel. Il ne limitait que les activités des éditeurs et des auteurs. *Stricto sensu*, les pauvres qui copiaient des livres à la main violaient probablement des copyrights. Mais personne ne les a jamais poursuivis, puisqu'on considérait le copyright comme un règlement industriel.

À l'âge de l'imprimerie, le copyright était facile à faire appliquer car il ne s'exerçait que chez les éditeurs – et par nature, un éditeur cherche à se faire connaître. Si l'on cherche à vendre des livres, il faut dire aux gens où

venir les acheter. Il n'était pas besoin d'aller faire respecter le copyright dans toutes les chaumières.

Enfin, le copyright fut peut-être bénéfique dans un tel contexte. Aux États-Unis d'Amérique, les spécialistes en droit considèrent le copyright comme un échange, un accord entre le public et les auteurs. Le public abandonne certains de ses droits naturels à la réalisation de copies, et reçoit en échange le bénéfice d'un plus grand nombre de livres créés et produits.

Est-ce un accord avantageux? Si tout un chacun ne peut réaliser de copies efficacement faute de disposer d'une presse à imprimer – et rares étaient les possesseurs de tels engins – il advient que le grand public abandonne une liberté qu'il ne peut exercer, une liberté impraticable. Si l'on a l'occasion d'échanger une potentialité inutile contre quelque chose de valeur, on réalise un marché gagnant. C'est pourquoi le copyright fut probablement un échange avantageux pour la société à cette époque.

Mais les temps et le contexte changent, et cela remet en question notre évaluation éthique du copyright. Les principes éthiques ne changent en rien avec les progrès techniques; ils sont trop fondamentaux pour être influencés par de telles contingences. Mais toute décision concernant un problème donné dépend des conséquences des choix disponibles, et les conséquences d'un choix donné dépendent du contexte. C'est ce qui se produit dans le domaine du droit du copyright, car l'âge de l'imprimerie prend fin et cède peu à peu la place à l'âge des réseaux informatiques.

Les réseaux informatiques et les techniques numériques de traitement de l'information nous ramènent dans une situation semblable aux temps anciens, où quiconque pouvant lire et utiliser l'information pouvait également la copier, et ce, aussi facilement que n'importe qui. Ce sont des copies parfaites, aussi bonnes que les copies qu'un autre réaliserait. La centralisation et les économies d'échelle introduites par l'imprimerie et les techniques apparentées ne s'appliquent donc plus.

Cette évolution du contexte modifie la manière dont le droit du

copyright fonctionne. Il ne fonctionne plus comme un règlement industriel, mais comme une restriction draconienne à l'encontre du public. C'était une restriction imposée aux éditeurs pour le bien des auteurs. C'est devenu, dans la pratique, une restriction imposée au public pour le bien des éditeurs. Le copyright n'était pas douloureux et ne soulevait pas de controverses, puisqu'il ne restreignait pas le public. Aujourd'hui, ce n'est plus le cas. La priorité numéro un des éditeurs est de confiner tout possesseur d'ordinateur. Le copyright était facile à faire appliquer en tant que restriction imposée aux éditeurs, car ils avaient pignon sur rue. C'est maintenant devenu une restriction portant sur tout un chacun. Son application implique surveillance, intrusions, et châtiments sévères, moyens dont nous voyons la mise en lois et en œuvre aux États-Unis d'Amérique et dans d'autres pays.

On pouvait défendre la thèse que le copyright était un marché avantageux pour le public, qui abandonnait des libertés qu'il ne pouvait exercer. Il peut désormais les exercer. Que faire si l'on découvre soudainement une utilité à une potentialité jusque là troquée faute de lui trouver un intérêt? On peut désormais la consommer, l'utiliser. Que faire? On n'y renonce plus, on en garde des parties. C'est l'impulsion naturelle. C'est pourquoi, à chaque occasion où il peut exprimer sa préférence, c'est ce que le public fait: il conserve une portion de cette liberté, et l'exerce. Napster en est un exemple frappant: le public décide d'exercer sa liberté de copier au lieu de l'abandonner. La tendance naturelle, pour faire correspondre le droit du copyright aux circonstances actuelles, est de réduire le pouvoir des détenteurs de copyright, réduire les restrictions qu'ils imposent au public, et augmenter les libertés que le public conserve.

Voilà qui déplaît aux éditeurs. Ils recherchent exactement l'inverse. Ils veulent développer le pouvoir du copyright jusqu'à contrôler fermement toute utilisation de l'information. Ces pressions ont fait voter des lois attribuant aux pouvoirs associés au copyright des extensions sans précédent.

Certaines des libertés accordées au public à l'âge de l'imprimerie lui sont désormais retirées.

Examinons par exemple le cas des livres électroniques. C'est un thème tellement à la mode qu'il est difficile d'y échapper. J'ai pris l'avion pour le Brésil et le magazine de bord contenait un article annonçant que d'ici 10 ou 20 ans, nous passerions tous aux livres électroniques. Voilà clairement une campagne financée par quelqu'un. Dans quel but? Je crois que j'ai deviné. Les livres électroniques sont l'occasion de retirer aux lecteurs des livres imprimés certaines des libertés qu'ils ont réussi à conserver – telles que la liberté de prêter un livre à un ami, de l'emprunter dans une bibliothèque publique, d'en vendre un exemplaire à un magasin de livres d'occasion, d'en acheter un exemplaire de manière anonyme, sans laisser de trace dans une quelconque base de données. Et, qui sait, le droit de le lire deux fois.

Voilà des libertés que les éditeurs souhaiteraient nous retirer, mais qui dans le cas des livres imprimés provoqueraient une levée de boucliers car ce serait une prise de pouvoir trop voyante. La stratégie indirecte qu'ils ont trouvée est donc la suivante: tout d'abord, on obtient de la loi qu'elle retire ces libertés aux livres électroniques à une époque où ils n'existent pas encore, ne provoquant ainsi aucune controverse. Il n'existe pas d'antériorité, d'utilisateurs de livres électroniques habitués à ces libertés et prêts à les défendre. Cette première étape fut atteinte avec le *Digital Millennium Copyright Act* en 1998. Ensuite, on introduit les livres électroniques et peu à peu on incite tout le monde à passer des livres imprimés aux livres électroniques. Finalement, le résultat est que les lecteurs ont perdu ces libertés sans qu'à aucun moment, ils ne s'en soient vu priver et aient donc eu l'occasion de se battre pour les conserver.

Pendant ce temps, on observe des remises en question des libertés de disposer d'autres types d'œuvres. Les films sur DVD, par exemple, sont publiés dans un format chiffré et secret – telle était initialement son intention en tout cas – et la seule ma-

nière d'obtenir des *majors* du cinéma la description du format, afin d'être capable de construire un lecteur de DVD, est de signer un contrat s'engageant à brider le lecteur. Ainsi, il est devenu impossible au public d'exercer les droits qui lui sont garantis par la loi. C'est alors que des programmeurs européens astucieux ont deviné le format des DVD et publié un logiciel libre capable de lire les DVD. Cela a permis aux utilisateurs du système GNU/Linux ayant acquis des DVD de regarder les films qu'ils avaient achetés – exigence parfaitement légitime au demeurant. On devrait pouvoir le faire avec des logiciels libres.

Mais les *majors* n'étaient pas de cet avis et elles ont porté l'affaire en justice. Voyez-vous, elles ont produit de nombreux films où un savant fou se voyait objecter «*Mais docteur, il y a des choses que l'Homme ne doit pas savoir.*» Il faut croire qu'elles ont trop regardé leurs propres films, car elles en sont venues à croire qu'il était mauvais pour l'Homme de connaître le format des DVD. Elles ont ainsi obtenu, par décision de justice, la censure complète de tout logiciel capable de lire les DVD. Il est même interdit désormais de tirer un lien vers un site qui propose cette information en toute légalité, hors des États-Unis d'Amérique. Il a été fait appel de cette décision, et je suis fier d'annoncer, bien que mon rôle reste mineur dans cette bataille, que j'ai signé un rapport technique sur le sujet.

Le gouvernement des États-Unis d'Amérique a pris ouvertement parti pour la partie adverse. Cela n'est pas surprenant quand on se rappelle la véritable raison de la mise en place du *Digital Millennium Copyright Act*: le système de financement des campagnes électorales, qui n'est rien d'autre qu'une corruption légale, où les grandes entreprises achètent les candidats avant même leur élection. Les élus ne connaissant que trop bien leurs maîtres – ils savent qui ils doivent vraiment servir – ils font passer les lois qui renforcent le pouvoir des grandes entreprises.

Nous ne savons pas ce qu'il adviendra de cette bataille en particulier. L'Australie a depuis voté une loi semblable, et l'Europe est sur le point

de le faire; l'intention est clairement de n'oublier aucun pays où cette information pourrait être mise à disposition du public. Mais ce sont les États-Unis d'Amérique qui demeurent champion du monde des entraves au grand public à distribuer les informations préalablement publiées.

Cependant, ce n'est pas le premier pays à faire de ce combat sa priorité. L'Union Soviétique y attachait également une grande importance. Les copies et distributions non autorisées y portaient le nom de *samizdat* et les Soviétiques avaient mis au point un certain nombre de méthodes pour les contrer. D'abord, des gardes surveillaient tous les copieurs et vérifiaient les documents que les gens copiaient, afin d'interdire toute copie illicite. Deuxièmement, de durs châtiements étaient réservés aux contrevenants: on pouvait être envoyé en Sibérie. Troisièmement, l'incitation à la dénonciation encourageait chacun à moucharder voisins ou collègues auprès de la police de l'information. Quatrièmement, la responsabilité collective: «*Toi! Tu surveilles ce groupe! Si je prends l'un d'entre eux à réaliser des copies interdites, tu vas en prison. Alors surveille-les bien.*» Et cinquièmement, la propagande qui, dès l'enfance, visait à convaincre tout le monde que seul un ignoble ennemi du peuple serait capable de se livrer à de telles activités.

Chacune de ces mesures est en place aux États-Unis d'Amérique en ce moment. D'abord, des gardes qui surveillent les copieurs. Dans les boîtes à copies, des gardes vérifient ce que vous copiez. Mais le coût de gardes pour vérifier ce que vous copiez avec votre ordinateur serait prohibitif; le travail humain coûte trop cher. Alors, on utilise des robots. C'est le but du *Digital Millennium Copyright Act*. Ce logiciel infiltre l'ordinateur: c'est le seul portail qui vous permet d'accéder à certaines informations, et c'est lui qui vous empêche de réaliser des copies.

Il est prévu d'inclure ce logiciel sur tous les disques durs, de telle manière que certains fichiers de votre propre disque ne vous soient accessibles qu'après l'accord d'un serveur réseau.

Il serait illégal de contourner cette protection, ainsi que d'expliquer à d'autres comment la contourner.

Deuxièmement, les durs châtiements. Voici quelques années, il n'était pas illégal de copier des informations et de les donner à vos amis; cela n'avait jamais été illégal aux États-Unis d'Amérique. C'est depuis devenu un délit, de telle sorte qu'il est maintenant possible de passer des années en prison pour avoir partagé avec autrui.

Troisièmement, les informateurs. Vous avez peut-être vu les publicités à la télévision ou dans le métro de Boston: on incite désormais à balancer les collègues à la police de l'information, qui s'appelle officiellement *Software Publishers Association*.

Quatrièmement, la responsabilité collective. Aux États-Unis d'Amérique, ce but fut atteint en contraignant les fournisseurs d'accès à l'Internet et en les rendant responsables de tout ce que leurs clients publient. La seule manière pour eux de ne pas être tenus responsables est d'appliquer une procédure sans recours consistant à déconnecter ou détruire l'information qui fait l'objet d'un contentieux dans les deux semaines qui suivent la plainte. J'ai entendu qu'il y a quelques jours un site de protestation intelligent mettant en cause la *City Bank* fut ainsi déconnecté. De nos jours, la présomption d'innocence a vécu: on n'a plus le droit de présenter sa défense, et on se fait tout simplement supprimer le site Web.

Enfin, la propagande, dès l'enfance. C'est le but du mot *pirate*. Si vous vous souvenez, ce terme était formellement réservé, voici quelques années, aux éditeurs qui ne payaient pas l'auteur. Son sens a depuis complètement changé et ce mot désigne les membres du public qui échappent au contrôle de l'éditeur. On l'utilise pour convaincre les gens que seul un méchant ennemi du peuple peut se livrer à des activités de copie interdite. Il sous-entend que partager avec son voisin est éthiquement équivalent à lancer un abordage. J'espère que vous refusez cet amalgame et si tel est le cas, j'espère que vous refuserez d'utiliser ce mot dans ce sens.

Ainsi, les éditeurs achètent des lois pour étendre et conforter leur pouvoir. De plus, ils allongent la durée du copyright. La constitution des États-Unis d'Amérique prévoit que l'exercice du copyright doit être limité dans le temps, mais les éditeurs souhaitent tout le contraire. Comme il serait toutefois difficile de faire amender la Constitution, ils ont trouvé une manière détournée d'obtenir le même résultat plus facilement. Tous les 20 ans, ils allongent de 20 ans la durée légale du copyright, avec effet rétroactif. C'est ainsi qu'à tout instant, la durée légale du copyright est finie, et tout copyright donné prendra théoriquement fin à un moment. Mais cette expiration ne sera jamais atteinte dans la pratique, puisque les copyrights sont allongés de 20 ans tous les 20 ans; plus aucune œuvre n'entrera donc jamais dans le domaine public. C'est le principe du copyright perpétuel.

La loi qui en 1998 a étendu de 20 ans la durée du copyright est connue sous le nom de Mickey Mouse Copyright Extension Act (loi d'extension de copyright Mickey), puisque l'un de ses plus virulents défenseurs était Disney. Les gens de Disney se sont rendu compte que le copyright portant sur Mickey allait expirer, événement qu'ils ne souhaitent absolument pas, ce copyright étant très lucratif.

J'étais censé vous parler de *copyright et mondialisation*. Si on étudie la mondialisation, on observe la mise en place d'un certain nombre de politiques au nom de l'efficacité économique ou de prétendus traités de libre échange – traités dont le véritable but est d'augmenter le pouvoir des grandes entreprises au détriment des lois et des politiques. Ce ne sont pas vraiment des traités de libre échange. Ce sont des traités qui mettent en place un transfert de pouvoir: ils retirent aux citoyens le pouvoir de décision – décisions portant sur leurs propres intérêts ou ceux de leur pays – pour l'attribuer à des entreprises qui ne prendront pas en compte les intérêts de ces citoyens.

Pour ces entreprises, la démocratie est un problème, et ces traités visent à le résoudre. Par exemple, l'ALÉNA prévoit des cas autorisant

des entreprises à attaquer en justice un gouvernement étranger pour annuler une loi qu'elles estiment interférer avec leurs profits dans ce pays étranger. C'est ainsi que les entreprises étrangères ont plus de pouvoir que les citoyens du pays.

On observe des tentatives d'étendre ce système au-delà des limites prévues par l'ALÉNA. C'est par exemple l'un des buts de la prétendue zone de libre échange des Amériques que d'étendre ce principe à tous les pays d'Amérique du Sud et des Caraïbes, et les accords multilatéraux sur les investissements avaient pour but d'étendre ce principe au monde entier.

Les traités signés dans les années 1990 ont commencé à imposer le copyright au monde entier, d'une manière plus forte et plus restrictive qu'auparavant. Ce ne sont pas des traités de libre échange. Ce sont des traités de commerce contrôlés par les entreprises et dont le but est d'attribuer aux entreprises le contrôle du commerce mondial, afin d'éliminer tout libre échange.

Au XIX^e siècle, les États-Unis d'Amérique, alors un tout jeune pays, ne reconnaissaient pas les copyrights étrangers. Ce fut une décision prise après mûre réflexion, et c'était une bonne décision. Les débats ont conclu que la reconnaissance par les États-Unis d'Amérique des copyrights étrangers ne serait pas avantageuse, et qu'elle aurait pour effet de vider les caisses sans rien apporter en échange.

On pourrait de nos jours appliquer ce raisonnement aux pays en voie de développement, s'ils n'étaient pas contraints par la puissance de ces mêmes États-Unis d'Amérique d'aller contre leurs intérêts. Dans un tel contexte, c'est en fait une erreur de parler d'intérêts des pays. Je suis sûr que vous avez pour la plupart entendu parler du raisonnement vicié qui tente de mesurer l'intérêt public en ajoutant toutes les richesses individuelles. Si les États-Uniens actifs perdaient un milliard de dollars et que Bill Gates gagnait deux milliards de dollars, cela profiterait-il à la population des États-Unis d'Amérique, de manière générale? Cela profiterait-il aux États-Unis d'Amérique? Si l'on ne prend que le

total en compte, cela semble une bonne affaire. Cet exemple montre cependant que le seul examen du total ne suffit pas à porter un jugement parce que Bill Gates n'a pas vraiment besoin de deux milliards de dollars supplémentaires, alors que la perte d'un milliard de dollars serait douloureuse pour une population qui ne jouit pas d'un patrimoine comparable. Et bien, dans toute discussion sur ces traités de commerce, quand on entend parler des intérêts de tel ou tel pays, c'est le même calcul qui est fait: en réalité c'est du total des revenus particuliers au sein de chaque pays dont il est question. On regroupe les riches et les pauvres dans le même calcul. Il s'agit donc d'un prétexte pour appliquer ce même raisonnement vicié afin de vous faire oublier la distribution des richesses au sein du pays et d'éviter que vous vous demandiez si le traité creusera les fossés et rendra cette distribution encore plus inégale – comme ce fut le cas aux États-Unis d'Amérique.

Ce n'est pas l'intérêt des États-Unis d'Amérique qu'on sert en faisant respecter le copyright partout dans le monde. C'est l'intérêt de certains propriétaires et patrons, dont beaucoup se trouvent aux États-Unis d'Amérique. En aucun cas, cela ne sert l'intérêt public.

Que faudrait-il faire, dans ce cas? Si l'on croit à la justification et au but du copyright tel que précisés dans la constitution des États-Unis d'Amérique – la promotion du progrès –, quelles seraient des politiques intelligentes à mettre en place à l'âge des réseaux informatiques? Clairement, au lieu d'augmenter les pouvoirs de copyright, il nous faut les limiter ou les diminuer, afin d'offrir au grand public un domaine de liberté où les gens pourraient faire usage des bénéfices des techniques numériques et des réseaux informatiques. Jusqu'où aller? C'est une question intéressante, car je ne crois pas qu'il faille abolir complètement le copyright. L'idée de troquer certaines libertés en échange de progrès peut encore être avantageuse à certains niveaux, même si l'exercice traditionnel du copyright abandonne trop de libertés. Si l'on veut réfléchir intelligemment à cette question, il nous faut commencer par reconnaître qu'il n'y a aucune raison de vou-

loir tout uniformiser. Il n'y a aucune raison de vouloir aboutir aux mêmes accords et aux mêmes compromis pour tous les types d'œuvres.

En réalité, ce n'est déjà pas le cas, car de nombreuses exceptions couvrent le domaine des œuvres musicales. Les lois de copyright traitent la musique très différemment. Mais c'est intelligemment que les éditeurs insistent – arbitrairement – sur l'uniformisation: ils choisissent un cas précis et arguent que dans ce cas particulier, il serait avantageux d'appliquer telle disposition en matière de copyright. Ils expliquent ensuite que par souci d'uniformité, il faut que tout soit régulé par les mêmes dispositions de copyright. Bien sûr, ils choisissent le cas de figure qui leur est le plus favorable, même s'il est rare et pas vraiment très important dans la pratique. Peut-être bien que ces dispositions sont les meilleures dans ce cas particulier. Il n'est nul besoin de payer le même prix pour tous les achats que l'on réalise: mille dollars pour une voiture neuve serait sans doute une excellente affaire, alors que le même prix pour une brique de lait serait une affaire exécrationnelle. On n'accepte pas de payer le même prix pour tout ce que l'on achète par ailleurs – pourquoi devrait-on l'accepter ici?

Il nous faut donc distinguer différents types de travaux, et je souhaite vous faire une proposition de classification.

Concernant les recettes, les programmes d'ordinateurs, les manuels et livres de cours, et les œuvres de référence telles que les dictionnaires et les encyclopédies: pour tous ces travaux fonctionnels, je crois que les problèmes qui se posent sont les mêmes que dans le cas du logiciel, et que par conséquent les mêmes conclusions s'appliquent. Les gens devraient même avoir la liberté d'en publier une version modifiée, car il est très utile de modifier les œuvres fonctionnelles. Tout le monde n'a pas les mêmes besoins: si j'ai écrit tel texte pour répondre à une nécessité ou à un souhait personnel, il ne répondra pas forcément à vos besoins, légèrement différents. Vous pourrez donc souhaiter modifier mon travail pour le rendre plus adéquat. D'autres pourront alors bénéficier de votre version modifiée,

car leurs habitudes seront similaires aux vôtres. Cela fait des siècles que tous les cuisiniers savent cela. Il est normal de donner ses recettes à d'autres, et il est normal d'adapter une recette. Si vous changez une recette et que les amis à qui vous servez le fruit de votre labeur aiment le résultat, ils vous demanderont probablement la recette. Alors peut-être leur donnerez-vous des copies de votre version. C'est exactement ce que nous fîmes, bien plus tard, dans la communauté du logiciel libre.

Voilà pour la première catégorie de travaux. La deuxième catégorie concerne les œuvres visant à rendre compte de la pensée et des idées de certains: le but de ces œuvres est de parler de leurs auteurs. Cette catégorie embrasse, entre autres, les mémoires, les essais d'opinion, les articles scientifiques, les petites annonces, les catalogues de biens à vendre. Toutes ces œuvres ont pour but de communiquer ce qu'une personne pense, a vu, ou croit. Les modifier serait trahir les auteurs; c'est pourquoi modifier ces œuvres n'est pas une activité utile socialement. Par conséquent, seule la copie exacte, ou *verbatim*, des ces œuvres doit être autorisée.

Il est alors légitime de se demander s'il faut autoriser une activité commerciale autour de telles copies *verbatim*, ou si la diffusion à but non lucratif suffit. On peut donc distinguer deux activités différentes, afin de les étudier séparément pour chercher des réponses à ces questions – le droit de faire des copies *verbatim* dans un but non lucratif, et le droit de faire des copies *verbatim* dans un contexte commercial. C'est un bon compromis que d'appliquer le copyright aux copies *verbatim* dans un contexte commercial, et d'autoriser sans autre contrainte les copies exactes exécutées et diffusées dans un but non lucratif. De cette manière, le copyright sur les copies *verbatim*, ainsi que sur toutes les versions modifiées – seul l'auteur pourrait approuver une version modifiée – fournirait une source de revenus de la même manière qu'il finance actuellement la création de ces œuvres, et dans les mêmes limites.

Autoriser les copies exactes à but non lucratif signifie que le copyright

ne doit plus s'immiscer dans la vie privée ni s'ingérer dans le foyer de chacun. Il redevient un règlement industriel, d'application facile et indolore, ne nécessitant plus de châtiments draconiens ni d'informateurs, sous le seul prétexte de le faire respecter. On obtient alors la majeure partie des bénéfices – et on évite le plus gros des horreurs – du système actuel.

La troisième catégorie contient les travaux esthétiques ou de loisirs, où le plus important est la sensation ressentie en regardant l'œuvre. La question de la modification est ici très difficile: d'un côté on trouve l'idée que ces œuvres reflètent la vision d'un artiste, et que toute modification viole et brise cette vision. D'autre part, il ne faut pas oublier les processus populaires, où des modifications apportées en chaîne par de nombreuses personnes produisent parfois un résultat extrêmement riche. Même si ce sont des artistes qui créent, les emprunts à des travaux précédents sont parfois fort utiles. Certaines pièces de Shakespeare se sont inspirées d'œuvres antérieures. Si les lois de copyright aujourd'hui en vigueur avaient alors été appliquées, de telles pièces auraient été illégales. C'est donc une question difficile que de décider comment réguler la publication de versions modifiées d'œuvres esthétiques ou artistiques, et il nous faudra peut-être subdiviser cette troisième catégorie pour résoudre ce problème. Par exemple, on peut imaginer de traiter les scénarios de jeux vidéo d'une manière; peut-être est-il souhaitable que quiconque puisse en publier des versions modifiées. Mais peut-être qu'un roman doit être traité différemment; peut-être que toute publication commerciale devrait faire l'objet d'un accord avec l'auteur.

Si la publication commerciale de ces œuvres esthétiques est couverte par le copyright, la plus grande part des revenus actuellement perçus servira encore à financer les auteurs et les musiciens, dans la mesure restreinte où le présent système les soutient, car il est très mauvais de ce point de vue. Cela pourrait donc être un compromis raisonnable, de la même manière que celui que nous avons imaginé pour les œuvres de la deuxième caté-

gorie, les œuvres qui représentent certaines personnes.

Si nous nous projetons dans le futur, à l'âge où les réseaux auront véritablement déployé leurs ailes, et où nous aurons dépassé l'état transitoire actuel, on peut imaginer une autre manière de rémunérer les auteurs pour leurs travaux. Imaginez un système de micro-paiements numériques qui vous permette d'être rémunéré pour votre travail. Imaginez un système de micro-paiements numériques qui permette d'envoyer de l'argent par l'Internet: il existe plusieurs manières d'obtenir un tel résultat en utilisant, par exemple, des procédés cryptographiques de chiffrement. Imaginez encore que les copies exactes de toutes ces œuvres esthétiques soient autorisées, mais que ces œuvres soit écrites de telle manière que toute représentation ou lecture ferait apparaître sur le côté de l'écran une boîte de dialogue disant «*Cliquez ici pour envoyer un dollar à l'auteur*», ou au musicien, ou au titre que la personne concernée se sera choisi. Cette boîte se trouverait là, sur le côté, sans interférer ni déranger, mais elle vous rappellerait que c'est une bonne idée d'encourager et de financer les auteurs et les musiciens.

Si vous aimez l'œuvre que vous lisez ou écoutez, alors vous vous direz tôt ou tard «*Et pourquoi n'enverrais-je pas un dollar aux auteurs? Un dollar, qu'est-ce pour moi? Je ne verrai pas la différence.*» Et les gens commenceront à envoyer des dollars. Le point positif: c'est que cela fait, des copies, l'allié des auteurs et des musiciens: toute copie envoyée par courrier électronique à un ami pourra inciter cet ami, lui aussi, à envoyer un dollar aux auteurs. Si vous aimez vraiment l'œuvre, vous pourrez envoyer plusieurs dollars – et ce dollar représente plus que ce que les auteurs perçoivent de nos jours si on achète un livre ou un CD, car seule une infime portion du prix de vente leur parvient.

Ces mêmes éditeurs, qui exigent un pouvoir et un contrôle total sur le public au nom des auteurs et des musiciens, paient ces derniers au lance-pierres.

Je vous recommande l'article de Courtney Love, publié dans le maga-

zine **Salon**[10], et traitant des pirates qui envisagent d'utiliser le travail des musiciens sans rémunérer ces derniers. Ces pirates sont les *majors* du disque, qui reversent en moyenne 4% des ventes aux musiciens. Bien sûr, les musiciens à succès touchent davantage. Ils reçoivent plus de 4% du faramineux chiffre d'affaires qu'ils suscitent – ce qui signifie que la grande majorité des musiciens sous contrat avec les *majors* du disque perçoivent moins de 4% de leurs ventes.

Voici comment le système fonctionne: la *major* investit dans une campagne publicitaire et considère ces frais comme une avance aux musiciens, alors même que les musiciens n'en perçoivent rien. Ainsi, ce n'est qu'en théorie qu'une certaine fraction du prix de vente de tout CD revient aux musiciens. En réalité, cette fraction rembourse les frais de la campagne publicitaire, et seuls les musiciens qui vendent beaucoup de disques commencent à toucher des royalties au-delà d'un certain seuil de ventes.

Bien sûr, les musiciens ne signent de tels contrats que dans l'espoir de faire partie des rares chanceux qui auront du succès. C'est en quelque sorte une loterie qu'on leur propose, une tentation. Même s'ils sont bons musicalement, ils peuvent être moins bons en perspicacité et en logique, et ne pas remarquer ce piège. C'est ainsi qu'ils en viennent à signer – et à n'en retirer qu'une campagne publicitaire. Bien, mais pourquoi ne pas leur offrir un autre type de campagne publicitaire, sans passer par un système restrictif et contraignant, un système industriel qui nous bassine de soupe facile à vendre. Pourquoi les musiciens ne s'allieraient-ils pas plutôt la tendance naturelle des auditeurs à partager les morceaux qu'ils aiment? Avec l'apparition d'une boîte de dialogue incitant à envoyer un dollar aux musiciens, les réseaux informatiques deviendraient leur vecteur de publicité – publicité qui pour l'instant est le seul bénéfice qu'ils retirent de leurs contrats avec les *majors* du disque.

Il nous faut reconnaître que le système actuel du copyright ne finance les musiciens que de manière exécrationnelle – aussi exécrationnelle que la manière

dont le commerce international élève le niveau de vie aux Philippines ou en Chine. Voyez ces zones franches où tout le monde travaille dans des ateliers de confection, et où tout y est produit. La mondialisation est un moyen très inefficace d'améliorer le niveau de vie des populations des pays pauvres. Soit un États-Unien payé 20 dollars de l'heure pour un travail; si on donne ce travail à un Mexicain qui l'effectue, disons, pour 6 dollars par jour, le résultat est qu'on a pris beaucoup d'argent au travailleur États-Unien, pour n'en reverser qu'une infime portion (quelques pourcents) au travailleur Mexicain – l'essentiel du bénéfice est pour l'entreprise. Si votre but est d'élever le niveau de vie au Mexique, voilà une bien mauvaise manière de procéder.

Il est intéressant d'observer le même phénomène et la même idée générale dans l'industrie du copyright. Au nom de ces travailleurs qui méritent sans aucun doute de recevoir quelque chose, on propose des mesures qui ne leur reversent qu'une infime portion des bénéfices et qui ont pour effet principal d'étayer le pouvoir qu'exercent les grandes corporations sur nos vies.

Quand on essaie de remplacer un très bon système, il faut travailler dur pour proposer une solution meilleure encore. Quand on sait que le système actuel est bancal, il n'est pas si difficile que cela de trouver une meilleure solution; de nos jours, la barre n'est pas placée très haut. Il faut toujours garder cela en tête quand on réfléchit aux politiques à appliquer en matière de copyright.

Je pense avoir dit tout ce que j'avais à dire. J'aimerais mentionner que demain (le 20 avril 2001) est, au Canada, le jour de l'opération «*appeler son employeur en prétendant être malade*» [11]. C'est en effet le début d'un sommet qui doit finaliser les négociations de la zone de libre échange aux Amériques pour tenter d'étendre le pouvoir des grandes entreprises à d'autres pays encore, et une grande manifestation de protestation est prévue au Québec. On a pu observer des méthodes extrêmes pour écraser cette protestation: de nombreux États-

Uniens sont bloqués à la frontière alors qu'ils sont censés pouvoir à tout instant entrer librement au Canada. On a construit, sur un prétexte complètement bidon, un mur-forteresse autour du centre-ville de Québec pour en exclure les protestataires. On a utilisé de nombreuses autres sales méthodes pour contrer la contestation publique de ces traités. S'il nous restait une once de démocratie après le transfert de pouvoirs, ôtés à des gouverneurs démocratiquement élus, et attribués à des entreprises et à des mul-

tionales nullement élues, peut-être bien que maintenant qu'on supprime et opprime les manifestations de protestation, cette infime portion de démocratie qui nous restait ne survivra pas.

J'ai consacré 17 ans de ma vie au logiciel libre et à des causes proches. Je ne l'ai pas fait parce que je pensais que c'était la cause politique la plus importante, mais parce que c'était le domaine où je voyais que je pouvais utiliser mon talent pour faire beau-

coup de bien. Mais les causes politiques ont évolué, et de nos jours la cause politique majeure est de résister à la tendance de donner le pouvoir aux entreprises, au détriment des peuples et des gouvernements. Je vois le logiciel libre et les questions apparentées concernant d'autres types d'informations, dont j'ai parlé aujourd'hui, comme un volet de cette cause fondamentale. C'est donc indirectement que j'ai travaillé pour cette cause. J'espère que ma contribution à cet effort n'est pas vaine.



RÉPONSE

David Thorburn:

Nous allons accepter les questions et les commentaires du public dans un instant. Je souhaite d'abord apporter une brève réponse générale. Il me semble que les réflexions de Stallman comportent deux éléments clefs, particulièrement forts et importants. Le premier est de reconnaître que les anciennes hypothèses et les vieux usages en matière de copyright ne sont plus appropriés, et sont remis en cause ou sapés par l'avènement de l'ordinateur et des réseaux informatiques. C'est peut-être évident, mais c'est essentiel.

Le deuxième est de comprendre que l'ère numérique nous force à réfléchir à la manière dont on distingue et dont on évalue les différentes formes de travaux intellectuels et créatifs. Stallman a certainement raison de signaler que certaines entreprises intellectuelles justifient plus que d'autres une protection accrue par le copyright. Tenter d'identifier de manière systématique ces différents types ou niveaux de protection par le copyright me semble une manière valable de traiter les problèmes posés aux travaux intellectuels par la venue de l'ordinateur.

Mais je pense détecter un thème supplémentaire, sous-jacent dans le discours de Stallman, un thème pas vraiment lié aux ordinateurs, mais plus largement aux questions d'autorité démocratique et au pouvoir que les gouvernements et les grandes entreprises exercent de façon croissante sur nos vies. Ce côté populiste et anti-mondialisation du discours de Stallman est stimulant mais aussi réducteur, voire simpliste. Et il est peut-être par trop idéaliste. Par exemple, comment un romancier, un poète, un parolier, un musicien, ou l'auteur d'un manuel universitaire survivraient-ils dans ce meilleur des mondes, où l'on encourage les gens à rémunérer les auteurs, sans les y obliger. En d'autres mots, il me semble que le fossé séparant ce qui se pratique actuellement et les possibilités visionnaires sur lesquelles Stallman spéculé est toujours immense.

Je conclurai donc en demandant à Stallman de développer certains aspects de son discours, et en particulier s'il a des idées supplémentaires sur la manière dont ceux que nous appellerons *les créateurs traditionnels* seraient protégés dans le système de copyright qu'il propose.

Richard Stallman:

Tout d'abord, je dois préciser un point: il ne faut pas utiliser le terme *protection* quand on se réfère à l'action du copyright. Le copyright restreint les gens. Le mot *protection* est un terme de propagande utilisé par les entreprises propriétaires de copyrights. *Protection* signifie empêcher une forme de destruction de quelque chose. Je ne pense pas qu'une chanson soit détruite si un plus grand nombre de copies de cette chanson sont diffusées plus souvent, auprès d'un plus grand nombre d'auditeurs. Je ne pense pas non plus qu'un roman soit détruit si un plus grand nombre de lecteurs en lisent des copies. Aussi n'utiliserai-je pas ce mot. Je pense qu'il incite à s'identifier avec le mauvais bord.

De plus, c'est une très mauvaise idée de réfléchir à la propriété intellectuelle, et ce, pour deux raisons: d'abord, cela présuppose une réponse à la question la plus fondamentale de ce domaine, c'est-à-dire: «*Comment ces choses doivent-elles être traitées, doivent-elles être traitées en tant qu'une forme de propriété?*» Utiliser le terme *propriété intellectuelle* pour décrire ce domaine revient à présupposer qu'on

répond affirmativement à cette question, que c'est la bonne manière d'envisager les choses, et qu'aucune autre n'est acceptable.

Ensuite, cela incite à généraliser abusivement. Propriété intellectuelle est un terme générique pour plusieurs systèmes juridiques différents, d'origines indépendantes, tels que le copyright, les brevets, les marques déposées, les secrets industriels, et d'autres choses encore. Ce sont là des domaines presque totalement différents: ils n'ont rien en commun. Mais ceux qui entendent le terme *propriété intellectuelle* sont incités à penser incorrectement, et ils imaginent que c'est un principe générique de la propriété intellectuelle qu'on applique à des cas particuliers, aussi supposent-ils que ces divers domaines juridiques sont similaires. Cela a pour conséquences, non seulement de rendre plus confuses les réflexions sur les bonnes manières de procéder, mais aussi d'empêcher de bien comprendre la loi car on suppose que les lois portant sur les copyrights, sur les brevets et sur les marques déposées sont semblables quand en réalité elles diffèrent totalement.

C'est pourquoi, si vous souhaitez encourager une réflexion prudente et une compréhension claire de la loi, il vous faut éviter le terme propriété intellectuelle. Parlez de copyrights, parlez de brevets, parlez de marques déposées ou parlez de quoi que ce soit d'autre dont vous voulez parler. Mais ne parlez pas de propriété intellectuelle. Toute opinion portant sur la propriété intellectuelle est presque forcément erronée. Je n'ai aucune opinion sur la propriété intellectuelle. J'ai des opinions sur les copyrights, sur les brevets, et sur les marques déposées, mais ce sont des opinions différentes. Je me suis forgé ces opinions à travers des chemine-ments de pensée différents, car ce sont des systèmes juridiques complètement différents.

Je me suis permis cette digression parce que je pense que c'est quelque chose d'extrêmement important.

Je vais maintenant répondre à votre question. Bien sûr, nous ne voyons pas bien, actuellement, de quelle manière cela fonctionnerait, si même cela pourrait fonctionner, de demander aux

gens de rétribuer volontairement les auteurs et les musiciens qu'ils apprécieraient. Il est évident toutefois que ce système fonctionnera d'autant mieux que de nombreuses personnes feront partie du réseau – et vous savez que ce nombre décuplera d'ici quelques années. Si nous essayions ce système aujourd'hui, il échouerait peut-être, et cela ne prouverait rien, car il pourrait fort bien fonctionner avec dix fois plus de participants.

D'autre part, nous ne disposons pas encore de ce système de micro-paiement numérique; aussi ne pouvons-nous pas vraiment mettre cela en pratique de nos jours. Tout au plus pouvons-nous essayer des schémas semblables. On peut s'inscrire à des services où on peut envoyer de l'argent à quelqu'un – des services tels que Pay Pal. Mais avant d'être capable d'envoyer de l'argent à quelqu'un à travers Pay Pal, il faut se farcir tout un galimatias, communiquer des informations personnelles, et le service conserve des traces des paiements effectués. Comment être sûr qu'ils n'abuseront pas de ces informations?

Même si l'envoi du dollar ne vous décourage pas, la complexité de la procédure le fera peut-être. L'idée générale est qu'il doit être simple comme bonjour de payer quand on en ressent l'envie, de telle sorte que seul le montant transmis pourrait décourager un éventuel mécène, et rien d'autre. Et si ce montant est suffisamment modeste, en quoi vous découragerait-il? Nous savons par ailleurs les proportions que l'adulation des idoles peut atteindre et nous savons que certains groupes ont encouragé leurs fans à copier et redifuser leur musique – groupes qui furent, ou sont encore, fort populaires, comme *Grateful Dead*. Ces groupes n'ont eu aucun problème à vivre de leur musique sous prétexte d'encourager leurs fans à en faire des enregistrements ni à échanger ces enregistrements. Ils n'ont même rien perdu en termes de ventes.

Peu à peu, nous quittons l'âge de l'imprimerie et entrons dans l'âge des réseaux informatiques, mais ce changement est graduel. Nombreux sont ceux qui continuent à acheter des disques, et cela continuera probablement

de nombreuses années encore, voire toujours. Tant que cela continuera, disposer de copyrights qui s'appliqueront sur les ventes de disques financera les musiciens à peu près aussi bien que de nos jours. Bien sûr, cela n'est pas très efficace – mais au moins, cela ne pourra pas empirer.

RÉFÉRENCES DU TRADUCTEUR

- [1] <http://www.stallman.org/>
- [2] <http://www.gnu.org/>
- [3] <http://web.mit.edu/m-i-t/forums/>
- [4] <http://web.mit.edu/m-i-t/forums/copyright/>
- [5] <http://www.gnu.org/philosophy/copyright-and-globalization.html>
- [6] <http://www.eff.org/>
- [7] http://www.eff.org/IP/DMCA/US_v_Sklyarov/us_v_sklyarov_fa.html
- [8] <http://www.eff.org/IP/P2P/Napster/>
- [9] <http://freenet.sf.net/>
- [10] <http://www.salon.com/tech/feature/2000/06/14/love/>
- [11] <http://www.rtmk.com/sick2000.html> ■

© Copyright 2001 Richard Stallman
[la reproduction exacte et la distribution intégrale de cet article sont permises sur n'importe quel support d'archivage, pourvu que cette notice soit préservée].

DES LOGICIELS À L'ŒIL

RICHARD TIMSIT, SIC, RICHARD.TIMSIT@EPFL.CH



Comme on me demande de rédiger l'intervention orale faite le 12 Juin lors de la rencontre-débat sur les Logiciels Libres à l'EPFL et que celle-ci n'était qu'une remarque construite dans le feu de la discussion, autant dire d'emblée qu'il me sera difficile de restituer ce bouillonnant contexte et d'être fidèle à mes propos.

Voulant réagir au fait que dans le débat tous les intervenants se posaient la question de la viabilité économique du modèle du logiciel libre sans se poser celle (indécente?) de la viabilité du modèle du logiciel propriétaire, j'ai voulu insister sur le fait que l'un n'avait rien à voir avec l'autre.

Ni du POINT DE VUE DE LA RAISON POUR LAQUELLE ON LES RÉALISAIT

Un logiciel propriétaire n'est qu'une marchandise destinée à des clients solvables tandis qu'un logiciel libre est conçu comme un outil devant être le plus utile possible pour soi et pour le plus grand nombre.

Ni du POINT DE VUE DE LA FAÇON DONT ON LES RÉALISAIT

Un logiciel propriétaire est écrit dans le secret d'une équipe avec des méthodes qui n'obéissent qu'à un souci de rentabilité tandis qu'un logiciel libre s'écrit sous le contrôle d'une communauté bien vigilante sur l'état de l'art en matière de génie logiciel.

Ni du POINT DE VUE DE LEUR PROMOTION

Le succès d'un logiciel propriétaire dépend surtout de la qualité du marketing tandis que celui d'un logiciel libre dépend du service qu'il peut rendre.

Ni du POINT DE VUE DE LEUR DEVENIR

Comme toute marchandise, un logiciel propriétaire destiné à être remplacé par une future version, est mort-né en quelque sorte, tandis qu'un logiciel libre – et c'est toute la force de la licence GPL – ne pourra que s'améliorer tant qu'un grand nombre d'utilisateurs en aura besoin, avec la garantie qu'il restera un Logiciel Libre.

Ces différences sont si importantes à mes yeux, que non seulement elles remettent en cause toute la façon dont on doit appréhender (et donc enseigner) la notion de qualité d'un logiciel – dans toutes les phases de son cycle de vie – mais aussi qu'elles nous interpellent profondément sur le plan éthique.

Peut-on encore accepter qu'un logiciel dont l'application est cruciale pour la société puisse s'appuyer sur du logiciel propriétaire dont on ne peut absolument rien garantir quant au fonctionnement ou à la pérennité?

Peut-on enseigner l'informatique dans une institution comme la nôtre en laissant toutes ces questions sous silence?

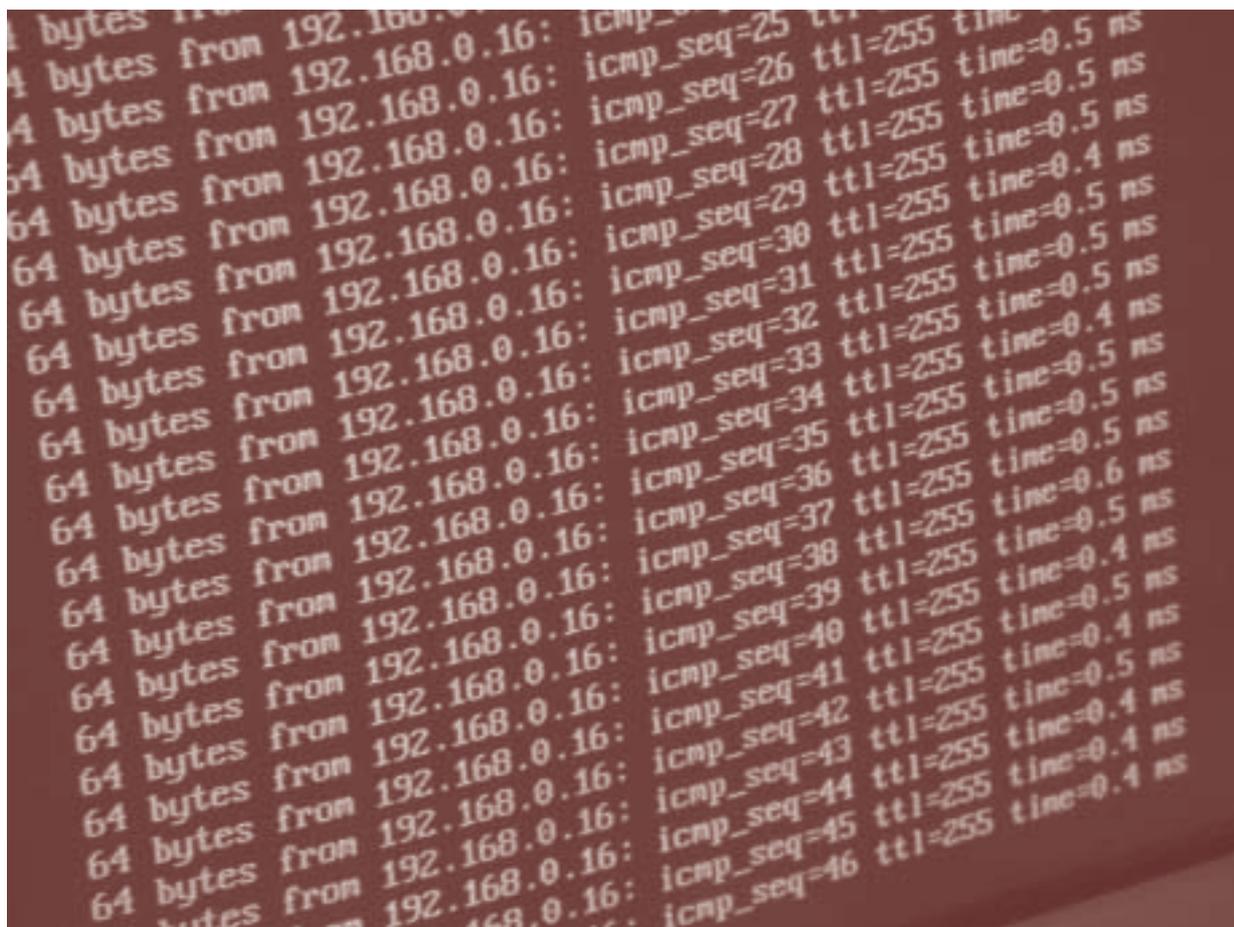
Peut-on avoir pris conscience de ce que notre place sur Internet doit au logiciel libre et se contenter d'en profiter?



ROBERTO di COSMO LORS DE LA RENCONTRE-DÉBAT SUR LES LOGICIELS LIBRES
DU 12 JUIN 2001 À L'EPFL

Pourquoi ces questions sont-elles si rarement abordées qu'elles donnent l'impression d'être subversives?

Que l'on puisse disposer de la plupart des logiciels libres gratuitement n'est pas la raison de ce titre. C'est le fait qu'on ne puisse désormais plus les perdre de vue qu'il faudrait retenir. ■



L'ENFANCE DE L'ART

UN PING QUI MARCHE C'EST DÉJÀ UNE PROMESSE DE COMMUNICATION. S'ABSTRAIRE DE CETTE SIGNIFICATION PREMIÈRE POUR Y VOIR UNE ÉVOCACTION NUMÉRIQUE DU CARRÉ BLANC SUR FOND NOIR, ET EN FAIRE UNE ILLUSTRATION, N'EST CE PAS LÀ UNE BELLE PROMESSE D'ÉMANCIPATION? CETTE IMAGE PROVIENT DE L'ALBUM DE PHOTOS DES RENCONTRE MONDIALE DU LOGICIEL LIBRE, BORDEAUX, JUILLET 2001.

<http://ism.abul.org/home.php3?langnew=fr>

Flash informatique

Les articles ne reflètent que l'opinion de leurs auteurs. Toute reproduction, même partielle, n'est autorisée qu'avec l'accord de la rédaction et des auteurs.

Rédacteur en chef:	Anne Possoz, fi@epfl.ch
Mise en page & graphisme:	Appoline Raposo de Barbosa
Comité de rédaction:	Jean-Daniel Bonjour, Jacques Bovay, Jean-Michel Chenais, Milan Crvcenin, Jacqueline Dousson, Jean-Jacques Dumont, Elaine Mc Murray, Martin Rajman, François Roulet, Christophe Salzmann & Jacques Virchaux
Impression:	Atelier de Reprographie EPFL
Tirage:	4000 exemplaires
Adresse Web:	http://sic.epfl.ch/publications/
Adresse:	SIC-SA EPFL, CP 121, CH-1015 - Lausanne
Téléphone:	+41 (21) 693 22 46 & 22 47
Date de parution:	4 septembre 2001



SOMMAIRE

2 Editorial

Anne Possoz

3 L'entreprise doit-elle adopter les logiciels libres?

Gilbert Robert & Frédéric Schütz

11 Intérêt général et propriété intellectuelle

Philippe Quéau

17 Logiciels Libres et Entreprises

Bernard Lang

23 FSF Europe – Liberté et Coopération

Frédéric Couchet

27 Les logiciels libres dans l'Education nationale française

Jean-Pierre Archambault

31 L'économie du Logiciel Libre : organisation coopérative et incitation à l'innovation

Dominique Foray & Jean-Benoît Zimmermann

41 Concours de la meilleure nouvelle: Denrée périssable

Adrien Bürki

45 Data mining avec R dans un monde libre

Diego Kuonen & Reinhard Furrer

51 GRASS GIS et R – main dans la main dans un monde libre

Reinhard Furrer & Diego Kuonen



57 Entretien avec Mélanie Clément-Fontaine à propos de la traduction de la GNU GPL

Loïc Dachary

59 Une Plate-forme d'Ingénierie Logicielle en Open Source

Opencascade

63 Copyright et Mondialisation à l'âge des réseaux informatiques

Richard Stallman

73 Des logiciels à l'œil

Richard Timsit



**L
O
G
I
C
I
E
L
S

L
I
B
R
E
S**



ISSN 1420-7192 771420 719001