Logiciels libres

Liberté, égalité, business

Jean-Paul Smets-Solanes

Benoît Faucon

Jean-Paul Smets-Solanes est ingénieur des mines et membre de l'AFUL (association francophone des utilisateurs de Linux et des logiciels libres). Il anime les sites www.smets.com (articles et études sur les NTIC), www.europe-inside.com (logiciels européens) et www.freepatents.org (brevets sur les logiciels).

Benoît Faucon est journaliste à l'hebdomadaire « Le Monde Informatique ».

Ont permis ou favorisé la réalisation de cet ouvrage par leurs remarques et leurs conseils avisés : Bernard Lang, Laurent W, Cyril Dhenin, Valérie Fageon, Stéfane Fermigier, Laurent Chemla, Ichiro Sekiba, Jacques Beigbeder, Jean-Pierre Laisné, Fabrice Rossi, Frédéric Couchet, Ken Sakamura, Roland Dyroff, Bob Young, Gael Duval, Henri S., Ross Jenkins, Dominique Robert et Damien Doligez. Nous les remercions vivement.

Toutes les marques citées dans cet ouvrage sont des marques déposées.

Rédaction et mise en page réalisées sous Linux avec Star Office.

Editeur : Patrick Altman

Edispher n'est lié à aucun constructeur ou éditeur.

Copyright Edispher, Paris 1999

71, rue du Commerce

75015 Paris

ISBN : 2-911 968-10-7

Licence chronodégradable

En application du droit d'auteur et en accord avec leur éditeur Edispher, Jean-Paul Smets et Benoît Faucon accordent au lecteur les droits d'usage suivants :

1. **Jusqu'au 1er avril 2000,** le lecteur est autorisé à copier ou adapter certaines parties de cet ouvrage (voir tableau ci-dessous) à condition de respecter le contrat de licence *Open Content* (voir Annexe F).
2. **A partir du 1er avril 2000**, le lecteur est autorisé à copier ou adapter l'ensemble de cet ouvrage (voir tableau ci-dessous) à condition de respecter le contrat de licence *Open Content* (voir Annexe F).

Jusqu'au 1er avril 2000	*A partir du 1er avril 2000*
5, 9 et Annexes en *Open Content*	Ensemble de l'ouvrage en *Open Content*

La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective », et, d'autre part, que les analyses et les courtes citations dans le but d'exemple ou d'illustration, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droits ou ayant cause, est illicite » (alinéa 1er de l'Article 40).

Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait une contrefaçon sanctionnée par les articles 425 du code pénal.

Table des matières

Préface.....1

Le phénomène Linux.....7

1. Définitions.....9

Ni freeware, ni shareware.....9
Ne peut être ni piraté ni racheté.....11
Un développement communautaire.....13

2. Histoire du logiciel libre.....15

La Free Software Foundation contre Bill Gates.....15
Et Linus créa Linux.....18
Le boom Internet.....21

3. En quoi peuvent-ils vous être utiles ?.....23

Linux, le système à tout faire.....23
Bureautique sous Linux.....37
Tout pour le Web.....42
Cas d'écoles.....44
Multimédia et création.....47
Bases de données.....49
PAO scientifique.....51
Programmation.....52
Sciences et ingénierie.....54
BSD, l'autre Unix libre.....55
eCos, l'embarqué pour tous.....59
Ce que l'on ne peut pas encore faire.....60

4. Chronique d'une adoption.....63

Apple fait son Unix.....63
Netscape, le retour aux sources.....64
Corel lance le Netwinder.....65
Adaptec.....66
IBM danse avec les fantômes.....66

La ruée des SGBD 67
Création et développement 68
Linux fait Java.....68
Les constructeurs brisent le tabou.....69
Les grands investissent le business du « libre ».....70
...Et Microsoft réagit.....71

5. Le droit du Logiciel Libre.....73
Le logiciel et le droit.....73
Les grands types de licences.....77
Conséquences.....83
Menaces légales sur le logiciel libre.....84

6. L'économie du logiciel libre.....91
La genèse des logiciels libres.....92
Un circuit de distribution structuré.....96
Des utilisateurs partenaires.....101
Les limites économiques du modèle libre.....108
Face aux modèles traditionnels.....115

7. L'informaticien citoyen.....119
Les laboratoires du freeware.....119
Le rôle des associations.....126

8. Des logiciels libres pour réguler le marché.....137
Les dérives de l'industrie du logiciel.....138
Echec aux stratégies de Winner Take All.....143
Pour une politique de la concurrence.....144

9. Une opportunité pour l'Europe.....153
Garantir l'interopérabilité.....160
Protéger les logiciels libres.....167

La nouvelle économie.....175

Glossaire.....179

Annexe A : Les essentiels.....183
Sites.....183
Textes fondateurs.....183
Articles de référence.....184
Les licences.....184

Annexe B : Carnet d'adresses.....185
La presse et l'information.....185
Où les télécharger ?.....185
Les éditeurs.....186
Acheter une distribution.....187
Les solutions.....188
Les organisations et associations.....188
Standardisation.....189
Sociétés.....189
Annexe C : Les solutions à base Linux.....193
Solution 1 : poste bureautique d'entreprise.....194
Solution 2 : Terminal X.....198
Solution 3 : Network Computer.....199
Solution 4 : station de création multimédia.....200
Solution 5 : station de travail scientifique.....203
Solution 6 : station de développement.....205
Solution 7 : firewall.....208
Solution 8 : réseau privé virtuel.....209
Solution 9 : serveur d'Intranet.....211
Solution 10 : serveur de fichiers.....213
Solution 11 : serveur de base de données.....215
Solution 12 : hébergement Web.....216
Annexe D : Les grands logiciels pour Linux.....219
Distributions.....220
Bureautique.....221
Clients Communication.....224
Serveur Communication.....225
Bases de données.....227
Sciences et ingénierie.....227
Graphisme.....228
Son et musique.....229
Jeux et simulation.....230
Emulation.....230
Développement.....231
Annexe E : Les partis pris.....235
Annexe F : Licence Open Content.....237
Index.....239

Préface

« Tout bien, toute entreprise, dont l'exploitation a ou acquiert les caractères d'un service public national ou d'un monopole de fait, doit devenir la propriété de la collectivité. »

Préambule de la Constitution du 27 octobre 1946
repris dans la Constitution du 3 juin 1958[1]

Le modèle de développement coopératif des logiciels libres a désormais fait ses preuves : innovation, fiabilité, sécurité, et performance au meilleur prix. Linux est devenu, en quelques années seulement, la première plate-forme d'origine européenne à atteindre aux Etats-Unis 17% du marché. Au moment même où Microsoft peine à produire une version de Windows NT/2000 qui réponde aux attentes des entreprises, Michael Cowpland, PDG de Corel et éditeur majeur de logiciels bureautiques, prédit que Linux sera d'ici cinq ans le système installé par défaut sur tous les serveurs.

1 http://www.elysee.fr/instit/c1946.htm et http://www.elysee.fr/instit/txt5814.htm

Mais Linux et les logiciels libres ne se réduisent pas au dernier avatar des luttes au sommet dans l'industrie du logiciel. Les logiciels libres annoncent un changement majeur de civilisation : l'avènement d'une société d'abondance.

Grâce à la mécanisation, la pénurie de biens pourrait bientôt disparaître dans les sociétés occidentales. Nous savons produire trop de nourriture, trop de voitures, trop de télévisions et d'autres biens essentiels ou futiles. En outre, les biens immatériels comme le cinéma, les logiciels ou la connaissance scientifique jouent un rôle croissant, déjà prépondérant, dans l'économie moderne. Avec l'essor d'Internet, ces biens tendent à perdre leur support matériel (film, papier; etc.) et leur localisation géographique. Ils acquièrent alors une caractéristique propre aux biens publics : leur valeur d'usage pour chacun n'est pas altérée par l'usage qu'en font les autres.

Cela signifie aussi qu'ils peuvent être reproduits indéfiniment à un coût pratiquement nul, c'est-à-dire sans travail humain ni utilisation de ressources matérielles. Il n'y a plus de relation entre le nombre d'unités produites et les coûts de production. Si vous me prenez ma voiture, j'en suis privé. Mais si vous recopiez ce texte, j'ai toujours ma propre copie et je ne m'en apercevrai même pas. La rareté n'existe pas naturellement dans un monde immatériel.

Mais avec la disparition de la rareté, la société telle que nous la connaissons voit son fonctionnement remis en question. L'économie politique classique a en effet pour principal objet la gestion de la pénurie. Lorsqu'il n'y avait pas assez de biens et de nourriture pour chacun, divers systèmes ont été imaginés pour structurer le flot des ressources afin d'assurer leur répartition dans la population. Et cela a permis d'avancer, tant bien que mal, en améliorant les techniques de production, sans pouvoir éviter la misère, les guerres et les révolutions.

Les situations changent souvent plus facilement que les schémas de

pensée. Aussi, la réponse trop souvent adoptée face à la surabondance dérangeante de biens immatériels, est d'inventer des mécanismes juridiques ou techniques pour réintroduire la rareté disparue. Les législations sur le droit d'auteur et les brevets furent à l'origine conçues pour défendre l'intérêt général en favorisant la création par la garantie d'une juste rétribution des créateurs. Dans le cas du logiciel, elles sont aujourd'hui dévoyées pour devenir une arme visant à interdire aux développeurs indépendants et aux PME l'accès à la création de produits innovants. Les coûts juridiques des extensions actuelles de la propriété intellectuelle aux idées et aux mécanismes immatériels les plus triviaux ont un effet dissuasif sur la majorité des inventeurs en puissance. De liberté fondamentale, la découverte des bonnes idées devient le privilège des puissants ou des fortunés.

En un temps où l'on prétend dynamiser le développement économique et technologique par l'effacement des contrôles et par l'élimination de l'intervention des Etats et de tout protectionnisme, pourquoi consacrer une telle énergie, tant de ressources et tant d'interventionnisme national et international, à établir des obstacles au libre usage des idées et des biens immatériels ? Alors même que trois mille ans d'histoire, et tout particulièrement celle de cette fin de siècle, montrent que c'est sur ces libertés que se fondent l'innovation, le progrès scientifique et technique, et la croissance économique. L'abondance de logiciels libres innovants, souvent de meilleure facture que les offres propriétaires du commerce, en est une nouvelle illustration.

Pour tirer le meilleur parti économique, social et humain des nouvelles dimensions immatérielles de notre monde, nous nous devons de les étudier et de les comprendre avec des yeux neufs, sans préjugés, en remettant en cause nos vieux modes de pensée et d'action, au lieu de les plaquer artificiellement sur un univers auquel ils sont étrangers.

Quelles sont les nouvelles voies à suivre ? Quelles sont les structures et les modes d'échange à adopter pour favoriser, pour le bien de tous, les

contributions de chacun ? Comment gratifier les auteurs de ces contributions ? Le monde du logiciel libre apporte quelques réponses à ces questions qui, au delà des logiciels, concernent la plus noble des activités humaines, la création.

Bernard Lang[2]

2 Bernard Lang est directeur de recherche à l'INRIA (Institut National de Recherche en Informatique et en Automatique) et secrétaire de l'AFUL (Association Francophone des Utilisateurs de Linux et des Logiciels Libres).

Pour ne pas subir

A Régis Allègre et à Cyril Dhenin

A Angela et au Baby Kamara

Le phénomène Linux

Soyons francs : écrire un livre sur les logiciels libres, c'est aussi songer au procès antitrust contre Microsoft. Les supporters de Linux, le système d'exploitation étendard de ce mouvement, sont souvent des pourfendeurs de ce monopole ; les éditeurs ralliés à Linux, sont souvent des concurrents de la firme de Redmond. Pourtant, il n'est rien de plus faux que de poser Windows et Linux en simples rivaux.

Il ne s'agit pas d'une alternative entre deux plates-formes, mais entre deux modèles informatiques et économiques qui peuvent d'ailleurs cohabiter : d'un côté, des entreprises ayant opté pour une offre propriétaire, qu'elles jugent nécessaire pour amortir leurs investissements. De l'autre des sociétés qui préfèrent donner leur produit et vendre des services. Ecrire sur les logiciels libres, ce n'est donc pas écrire contre Microsoft, mais à-côté.

De la même manière, écrire sur les logiciels libres, ce n'est pas nécessairement s'en faire les propagandistes. Mais plutôt apporter des réponses à toutes les questions que l'on peut se poser face à un modèle émergent, complexe et souvent mal compris. Cet ouvrage cherche donc à donner des clefs pour comprendre, réfléchir et agir à tous ceux que le phénomène Linux et les logiciels libres concerne dans leur vie profes-

sionnelle ou personnelle. Je suis utilisateur, comment m'en servir ? Je suis décideur, est-ce que cela constitue une solution viable pour mon entreprise ? Je suis fournisseur, quelle opportunité représente ce marché ?

Mais le logiciel libre, ce n'est pas qu'une question d'informatique. C'est aussi à tout citoyen que s'adresse ce livre. Derrière Linux, c'est en effet un nouveau modèle juridique, économique et social qui s'offre à nous. Et les auteurs sont convaincus que ce mode de fonctionnement a sa place dans notre société, ne serait-ce que pour faciliter l'accès de tous aux nouvelles technologies de l'information et pour favoriser la concurrence et l'innovation dans une industrie du logiciel sujette de façon récurrente aux abus de certains de ses acteurs.

1

Définitions

Parce qu'il est généralement gratuit ou bon marché, le logiciel libre est souvent confondu avec le *freeware* (logiciel gratuit) ou avec le *shareware* (logiciel partagé). Pourtant un logiciel libre n'est pas forcément gratuit. Ces différents types de produits correspondent à des principes très différents.

Ni *freeware*, ni *shareware*

Le *freeware* a explosé depuis quelques années[3]. Créé dans des conditions souvent artisanales par de jeunes développeurs, on en trouve des compilations sur des CD-ROM ou offertes avec des magazines informatiques. On peut aussi en télécharger à foison sur des sites spécialisés comme filez.com[4], tucows.com[5] ou download.com[6].

3 Cf. « Vive le logiciel libre » par Yves Eudes – Le Monde (18/03/96)
4 http://www.filez.com
5 http://www.tucows.com
6 http://www.download.com

Un *shareware* est un logiciel où l'utilisateur paie une contribution généralement modeste au concepteur du programme. Il s'agit donc d'un modèle de vente directe, du producteur au vendeur qui, en l'absence d'intermédiaire, permet de réduire les prix d'environ un facteur dix par rapport aux logiciels équivalents diffusés par le circuit de distribution traditionnel. Du fait de la faible pénétration du paiement électronique dans les pays du sud de l'Europe, ce modèle fonctionne relativement mal auprès des utilisateurs français. Mais les créateurs n'exigent pas toujours de l'argent, certains demandant des cartes postales ou… des caisses de bière en provenance du monde entier[7].

Ce qui distingue un logiciel libre d'un *shareware* ou d'un *freeware*, ce sont les droits bien plus importants que l'auteur du logiciel libre consent à l'utilisateur. Un logiciel libre, c'est avant tout :

- Un produit livré avec son code source. D'où le terme de logiciel « *Open Source* » (« à code source accessible à tous »), utilisé par Marc Andreseen, le dirigeant de Netscape à l'origine de l'adoption de ce modèle pour Communicator. C'est toute la différence entre Internet Explorer (logiciel gratuit depuis longtemps – mais pas libre) et Communicator (complètement gratuit depuis février 1998 mais aussi livré avec son code source dans le cadre du projet Mozilla[8]).
- Un contrat de licence qui autorise tout utilisateur à copier et à modifier le produit.
- Une communauté d'utilisateurs dont certains gèrent aussi le développement ou apportent parfois des améliorations au produit.

Les logiciels libres ne s'opposent donc pas aux logiciel commerciaux ; il existe d'ailleurs un nombre croissant de logiciels commerciaux et libres. Les logiciels libres s'opposent aux logiciels propriétaires, dont les auteurs

7 Id.
8 http://www.mozilla.org

interdisent la copie, l'adaptation et l'accès au code source. En ce sens, *freeware* et *shareware* sont des logiciels propriétaires.

Aussi, de nombreux auteurs de logiciels libres interdisent que les améliorations et les développements issus de leurs produits soient diffusés sous forme de logiciels propriétaires. En clair, comme chez Jean-Paul Sartre, la liberté de l'utilisateur de logiciels libres s'arrête là où commence celle des autres. Ce n'est en revanche pas le cas des logiciels du domaine public qui, contrairement aux logiciels libres, ne sont pas protégés par le droit d'auteur et peuvent donc aussi être diffusés sous forme de logiciels propriétaires.

Tableau 1. Comparaison des principaux types de logiciels

	shareware	*freeware*	*Logiciel libre*	*Domaine public*
Gratuit	Non	Oui	Souvent	Oui
Protégé par le droit d'auteur	Oui	Oui	Oui	Non
Peut-être copié librement	Non	Non	Oui	Oui
Code source disponible	Rarement	Rarement	Oui	Oui
Peut-être adapté librement	Non	Non	Oui	Oui
Peut-être revendu en tant que logiciel libre	Non	Non	Oui	Oui
Peut-être revendu en tant que logiciel propriétaire	Non	Non	Rarement	Oui

Ne peut être ni piraté ni racheté

Aux Etats-Unis, nombreuses sont les affaires de piratage qui ont mis aux prises, défenseurs des logiciels libres, autorités et éditeurs[9]. En 1990, lors de l'« *Operation Sundevil* », déclenchée par le FBI, plusieurs jeunes *hackers* furent arrêtés pour avoir divulgué des extraits du code source du système d'exploitation d'Apple. Par la suite, l'affaire La Macchia, du nom d'un étudiant du MIT qui avait offert des versions bêta de logiciels à télé-

9 Id.

charger sur son site, devait à nouveau faire rebondir le débat. Les éditeurs n'ayant pu prouver que l'accusé tirait un quelconque profit de la diffusion de ces copies, les poursuites furent abandonnées deux ans plus tard.

Pourtant, qui dit copie libre, ne veut pas dire piratage : si le système du logiciel libre fonctionne, c'est parce qu'il est régi par un cadre juridique strict : le principe du *copyleft* (gauche de copie), inventé par Richard Stallman, président de la Free Software Foundation (voir chapitre suivant). Ce terme s'oppose malicieusement à *copyright* (droit de copie) sous-entendu propriétaire : le *copyleft*, c'est le droit laissé à tous de recopier et de modifier un programme, voire de le distribuer à condition de conserver un code source accessible à tous.

La licence publique la plus couramment utilisée pour les logiciels libres est la GNU General Public Licence (GPL). La GPL donne le droit aux utilisateurs d'effectuer des améliorations sur un programme, de poster celles-ci sur Internet et même de revendre le fruit de ces développements. A une seule condition : livrer aussi le code source du nouveau logiciel sous licence GPL (voir chapitre 5).

Ce modèle s'adapte parfois assez mal aux entreprises qui souhaitent continuer à contrôler l'exploitation commerciale ou propriétaire de leurs produits. Ainsi, lorsqu'il a ouvert le code source de Communicator, Netscape a créé la Netscape Public Licence (NPL). Celle-ci lui permet d'intégrer les développements effectués sur la suite logicielle à son offre de serveurs sans pour autant avoir à étendre le *copyleft* à ces derniers.

Même chose pour la licence de BSD, l'Unix libre conçu à l'université de Berkeley. Cette licence permet de déposer un *copyright* propriétaire sur un produit issu des modifications de BSD, comme par exemple MacOS X, le nouveau système d'exploitation propriétaire d'Apple.

Mais certains défenseurs des logiciels libres émettent des réserves sur ce type de licences car elles cassent le cercle vertueux d'ouverture du

code et du développement par et pour l'utilisateur. D'autres défenseurs des logiciels libres estiment au contraire que toutes ces variétés de licences font partie d'un modèle plus général que celui développé initialement par la Free Software Foundation, un modèle mieux adapté à l'industrie et désormais adopté par elle sous le nom d' *Open Source Software* (logiciel à code source ouvert).

Le type de licence associé au droit d'auteur est donc l'élément clef qui permet le développement et l'amélioration constante des logiciels libres. Deux visions du droit d'auteur distinguent les deux modèles économiques actuellement en concurrence dans l'industrie du logiciel. D'un côté, les tenants du *copyright* propriétaire, qui estiment qu'il est nécessaire d'interdire la copie et l'accès au code source pour protéger leurs investissements en recherche et développement. De l'autre, les partisans du logiciel libre, qui considèrent que le code source est une propriété partagée et que la richesse se crée sur les services.

Un développement communautaire

Parce qu'il appartient à tous, le logiciel libre évolue de façon coopérative grâce aux modifications apportées par des développeurs à travers le monde entier. C'est le modèle décrit par Eric Raymond dans son article « The Cathedral and the Bazaar »[10] : en évitant l'appropriation du produit par qui que ce soit, le cadre juridique permet de fabriquer des logiciels que les utilisateurs peuvent maîtriser et faire évoluer de manière à ce qu'ils correspondent à leurs besoins.

Cette collaboration globale est la conséquence directe de la diffusion de l'Internet[11]. Grâce au réseau, il devient en effet facile de développer à plusieurs un logiciel. On trouvera toujours dix personnes à travers le

10 Eric Raymond « La Cathédrale et le Bazar » http://www.tuxedo.org/~esr/writings/cathedral-bazaar/
11 Cf. Id.

monde qui auront le même problème à résoudre. Le Net a aussi permis de faire passer à un coût zéro la diffusion d'un logiciel par son créateur. A l'origine, pour diffuser le logiciel dont on était l'auteur, il fallait presser un CD-ROM ou l'enregistrer sur des disquettes, puis les envoyer par la poste. Cela coûtait tellement cher que même les logiciels distribués conçus sans but lucratif étaient obligatoirement payants.

Figure 1. Eric Raymond

Aujourd'hui, diffuser sur Internet ne coûte rien à l'auteur, seul le temps de téléchargement étant à la charge de l'utilisateur. Du coup, le développement coopératif devient possible.

EN BREF – Le logiciel libre est souvent confondu avec le *freeware* (logiciel gratuit) ou avec le *shareware* (logiciel où l'on paie une contribution à l'auteur). Le logiciel libre est aussi assimilé à tort aux logiciels du domaine public. Pourtant, un logiciel libre, c'est avant tout un produit protégé par le droit d'auteur, livré avec son code source, régi par un contrat de licence publique qui autorise la copie et qui, grâce à Internet, est enrichi constamment par la communauté de ses utilisateurs.

2

Histoire du logiciel libre

La querelle du *copyleft* et du *copyright* met aux prises depuis ses débuts, les deux personnages qui donnent tout leur épice au débat, Richard Stallman, fondateur de la Free Software Foundation, et Bill Gates, créateur et patron de Microsoft. Avec à l'origine, un cadre unique : le campus de Cambridge à Boston, sur la côte Est des Etats-Unis[12]. Alors qu'en 1997, Bill Gates était, pour la plupart des analystes, sur le point de remporter une victoire totale et définitive, l'essor irrésistible de Linux sur la scène internationale semble donner en 1998 une nouvelle vie au modèle défendu depuis plus de vingt ans par Richard Stallman.

La Free Software Foundation contre Bill Gates

C'est au MIT (Massachussets Institute of Technology) que sont apparus, dans les années 50, les premiers ***hackers***. Mais loin de chercher à en-

12 Voir à ce sujet : « For the love of hacking » - Forbes 10/08/98

trer par effraction dans les bases de données du Pentagone (on parle plutôt à ce propos de *crackers*), les *hackers* faisaient dans le bricolage légal, cherchant avant tout à tirer parti du potentiel des gigantesques ordinateurs de l'université. C'est dans ce milieu fertile que naîtra plus tard la Free Software Foundation (FSF), à l'origine du mouvement des logiciels libres.

Dans les années 70, la communauté des ingénieurs et des chercheurs du MIT avait pris l'habitude de partager le code source de leurs programmes, considérant l'achat de la machine comme plus important que celui de la licence du logiciel. Arrivé en 1971 au laboratoire d'Intelligence Artificielle du célèbre institut, Richard Stallman, qui devait fonder plus tard la FSF, a donc été formé dans cette tradition. Ce qui ne semble pas avoir été le cas de deux autres étudiants de l'université voisine d'Harvard, Bill Gates et Paul Allen, futurs fondateurs de Microsoft.

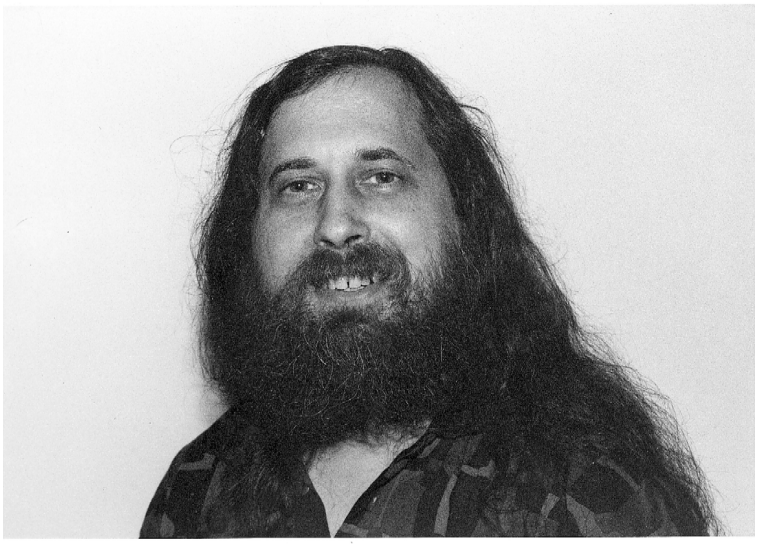

Figure 2. Richard Stallman

A quelques blocs du laboratoire de Stallman, les deux amis écrivirent en 1975 un système d'exploitation pour l'Altaïr 8800, une grosse machine

avec une rangée de boutons rouges pour tout système d'affichage et, déjà, un processeur Intel, le 8080. Conformément aux coutumes du lieu, quelqu'un copia les programmes des deux étudiants et les diffusa à travers le campus. A la grande colère de Bill Gates qui expédia aussitôt une « Lettre ouverte aux hobbyistes » à la presse étudiante locale. « La plupart d'entre vous volez vos logiciels. Ce faisant vous empêchez l'écriture de bons logiciels. Qui peut se permettre d'effectuer un travail professionnel pour rien ? », s'indignait le jeune homme. Plus tard, Bill Gates lancera MS Basic, en interdisant d'en effectuer une copie, même partielle ou destinée à un usage interne. Le modèle Microsoft, fondé sur le paiement de royalties et sur la vente de logiciels protégés par une licence interdisant la copie entre utilisateurs et le paiement de royalties, était né.

De son côté, sur le campus du MIT, Richard Stallman décida de reprendre le flambeau de la tradition universitaire. En 1984, il lança le projet GNU, ainsi choisi parce que c'était le nom le plus bizarre que Stallman ait trouvé dans le dictionnaire mais aussi parce qu'il s'agissait de l'acronyme résumant le mieux le programme fixé par son créateur : « *GNU's Not Unix* » (GNU n'est pas Unix).

A l'origine, le projet avait en effet pour but de créer un Unix libre de droit. Mais pour concevoir un tel logiciel, encore fallait-il disposer d'un environnement viable. Stallman s'assigna donc d'abord pour mission de créer des programmes fondamentaux pour le futur système d'exploitation (compilateurs, outils de développements, éditeurs de texte). L'année suivante, Stallman passa à la vitesse supérieure en fondant la Free Software Foundation (FSF), destinée à soutenir les développements de logiciels libres.

On lui doit la pièce juridique maîtresse du modèle des logiciels libre : la licence GPL ou « *GNU General Public License* », le contrat de licence sur lequel reposent la plupart des logiciels libres. Mais la FSF a aussi mis au point le débogueur gdb, le compilateur gcc, l'un des plus utilisés au monde, ou l'éditeur de texte Emacs, lui aussi très répandu chez les

VRAI OU FAUX : les partisans du logiciel libre sont uniquement des ennemis de Microsoft

Historiquement, faux. Dans les années 80, la Free Software Foundation (FSF) a appelé au boycott d'Apple pour avoir breveté et protégé par le copyright son système d'interfaces graphiques, ce qui empêchait notamment Microsoft d'utiliser ses icônes. Lors de l'opération Sundevil (voir chapitre 1), le constructeur du Macintosh lança d'ailleurs la police contre des étudiants qui avaient diffusé le code source de son OS.

Mais du fait du quasi-monopole de Windows sur le PC, les partisans des logiciels libres ont concentré leurs critiques contre Microsoft. Ils considèrent que la politique de *licensing* (octroi de licences) de l'éditeur restreint sérieusement la possibilité d'installer un autre OS que le sien dans les machines des grands constructeurs, tandis que l'usage de formats propriétaires fait des usagers de produits Microsoft une clientèle captive.

Il n'en reste pas moins que :

1. Le monde du logiciel libre n'est pas directement présent au procès de Microsoft mais seulement certains concurrents de la firme de Redmond (Sun, Nestcape, Apple, etc.).
2. Nombreux sont les utilisateurs de Linux qui n'ont rien contre l'utilisation d'applications conçues pour Windows : le créateur de Linux, Linus Torvalds, se sert de PowerPoint via une émulation Windows sur sa machine Linux.

Les partisans du logiciel libre sont davantage des apôtres de la « diversité des espèces » et de la liberté du logiciel que des croisés anti-Microsoft.

programmeurs ou les utilisateurs de systèmes de type Unix.

Et Linus créa Linux

Malgré le développement d'une chaîne complète de logiciels libres, il manquait encore à Stallman une pièce maîtresse : un système d'exploitation (OS) non propriétaire pour les faire tourner. L'un des grands projets de la Free Software Foundation était de concevoir un OS de type Unix en *copyleft*, baptisé Hurd. Hurd est, aujourd'hui encore, à un stade expérimental. Mais, entre temps, c'est un jeune Finlandais qui a réalisé le rêve

de Stallman. Son nom : Linus Torvalds.

Pourtant, c'est surtout le hasard qui a poussé Linus à concevoir ce qui constitue aujourd'hui, la principale alternative au monopole de Microsoft sur les systèmes d'exploitation pour micro-ordinateurs à processeur de type Intel.

Tout commence en 1990 lorsque Torvalds, alors étudiant en informatique, se retrouve à faire la queue pour accéder aux ordinateurs de la faculté d'Helsinski[13]. Surpopulation universitaire oblige, il n'y a qu'un terminal pour 16 utilisateurs. Du coup, Linus décide de s'acheter un PC. « Jusque-là, je n'en voulais pas parce que, sinon, j'aurais eu cette architecture minable avec ce système d'exploitation MS-DOS minable et je n'aurais rien appris », confiera-t-il plus tard au magazine Wired.

Figure 3. Linus Torvalds en famille

La solution viendra d'un système d'exploitation extrêmement basique mais livré avec son code source, Minix, un bébé Unix destiné à l'ensei-

13 The greatest OS that (n)ever was – Wired, août 1997 http://www.wired.com/wired/5.08/linux.htm

gnement[14]. Mais, les capacités de l'OS étant extrêmement limitées, Linus rajoute peu à peu des lignes de code pour naviguer sur les groupes de discussions, puis écrit un logiciel pilote de périphérique pour pouvoir utiliser un autre écran ou un nouveau clavier et un système de gestion de fichiers. Un peu plus tard, le Xfree86 Project[15] apportera une interface graphique au système d'exploitation : Linux, contraction humoristique du prénom de son créateur, Linus, et de Unix, le système dont il s'inspire, était prêt à conquérir le monde.

Mais Linux ne serait jamais venu au monde sans l'aide des centaines de développeurs qui ont participé, en communiquant sur le Web, à sa création. Ceux-ci mettent aussi à profit les travaux de leurs prédécesseurs de la FSF, intégrant autour du noyau de l'OS de nombreuses commandes issues du projet GNU ou utilisant les compilateurs (gcc) et les outils de développement conçus par la fondation. Il faudra attendre 1994 pour que Linux 1.0, première version vraiment aboutie de l'OS, ne soit disponible. Depuis, le nombre de lignes de code est passé de 10 000, lors des débuts héroïques, à plus d'un million.

Le développement coopératif garantit d'ailleurs la pérennité de l'OS, même si son créateur venait à s'en désintéresser. Fin 1996, Linus Torvalds annonça qu'il arrêtait de se consacrer à plein temps au développement de son bébé parce qu'il venait d'avoir une autre naissance, bien réelle (sa femme venait d'accoucher) et qu'il lui fallait gagner sa vie. A la grande surprise de la communauté Linux, l'étudiant rejoignit donc une *start up* de la Silicon Valley, Transmeta dont Paul Allen, cofondateur de Microsoft, est d'ailleurs l'un des principaux investisseurs.

Depuis, Linus a donc légèrement réduit son implication dans le projet même s'il en reste le coordinateur et est payé pour travailler dessus à temps partiel. Mais le développement continue plus vite que jamais. Sor-

14 Conçu par Andrew Tannenbaum, de l'université Vrije (Amsterdam), il était destiné à être utilisé avec son manuel « Operating Systems : Design and Implementation ».

15 Le Xfree86 Project a pour but la fourniture des versions libres du système X Window qui est aujourd'hui la norme pour la gestion des interfaces graphiques sur les systèmes de type Unix.

tie début 1999, la version 2.2 apporte d'ailleurs de nombreuses modifications, parmi lesquelles :

- la simplification du développement des pilotes de périphériques ;
- la reconnaissance automatique (*plug and play*) des périphériques PCI ;
- un meilleur support des configurations multiprocesseurs ;
- le prise en compte en standard des disques durs et partitions utilisant les systèmes de fichiers FAT32, NTFS ou HFS ;
- la gestion des périphériques USB et des DVD-ROM ;
- la réécriture complète du gestionnaire audio ;
- l'accès direct aux périphériques vidéo à partir du noyau (*frame buffer console*) ;
- une prise en compte étendue des protocoles Ethernet et RNIS, des modems câble et des modems analogiques à la norme V90.

La communauté Linux tend à se structurer avec la création d'un consortium, **Linux International**, chargé de promouvoir et de superviser le développement de l'OS, et de revendeurs à valeur ajoutée qui ont développé leur propres versions de Linux et vendent du service autour du système d'exploitation (SuSE, Red Hat, Caldera, Pacific High Tech, etc.).

Le boom Internet

La diffusion de l'usage des logiciels libres est indissociable du développement de l'Internet. D'abord, parce que celui-ci a provoqué une explosion du nombre d'utilisateurs potentiels, qui n'avaient plus qu'à aller télécharger des produits gratuits. Ces outils ont notamment intéressé le public des développeurs, des particuliers et des étudiants qui, comme à l'époque du micro-ordinateur Sinclair, souhaitaient écrire ou modifier des programmes. Qui plus est, avec l'arrivée de la plate-forme Linux, les ou-

tils GNU et les bibliothèques de logiciels développés par les universités ou issus de la recherche publique ont cessé d'être réservés aux utilisateurs de stations Unix très chères pour devenir accessibles à tous.

Mais surtout, l'infrastructure du réseau est elle-même gérée avec des outils à code source modifiable. Rien d'étonnant à cela puisque les universités et les fournisseurs d'accès Internet préfèrent généralement utiliser des logiciels conformes à ce modèle pour des questions de fiabilité ou de coûts. Le réseau mondial utilise donc couramment des logiciels libres pour : la transmission du courrier électronique (Sendmail), ses serveurs de fichiers FTP (WU-FTP), la gestion des noms de domaine (Bind), les *news* (Inn) et la programmation des pages Web (langage Perl).

L'absence d'offre commerciale mature dans le domaine du Web a aussi donné naissance en 1995 à Apache, aujourd'hui le serveur Web le plus utilisé. Apache a vu le jour lorsque l'un des responsables techniques du magazine Wired, Brian Behlendorf, s'est mis, avec quelques collègues, à faire des modifications ou *patches* (d'où le nom « Apache », prononcé « A-pa-tchi » en anglais) à partir de l'un des seuls serveurs libres disponible à l'époque : le HTTP *daemon* écrit par Rob McCool, pour le National Center for Supercomputing Application (NCSA) de l'Université d'Illinois. Grâce aux centaines de développeurs qui se sont joints à ce projet, Apache est aujourd'hui considéré comme le serveur Web le plus efficace et est utilisé par 52% des sites de la toile mondiale, selon Netcraft.

EN BREF – Les logiciels libres ont été initialement créés comme une alternative à l'informatique propriétaire avec la FSF de Richard Stallman. Ils ont fait leur entrée dans l'entreprise avec la diffusion du système d'exploitation Linux et le développement de l'Internet. Aujourd'hui, ils sont devenus la principale forme de concurrence aux éditeurs de logiciels propriétaires en position dominante.

3

En quoi peuvent-ils vous être utiles ?

Linux, le géant finlandais avec 4 à 10 millions d'utilisateurs dans le monde, est la partie émergée de l'iceberg. L'offre de logiciels libres couvre en effet des domaines qui vont de la PAO scientifique aux solutions bureautiques, en passant par les supercalculateurs et le domaine où ils règnent en maîtres : le Web. Le lecteur trouvera en annexe C un guide de solutions pour couvrir les besoins du particulier et de l'entreprise.

Linux, le système à tout faire

Linux a toutes les qualités d'un Unix, la gratuité et la compatibilité multi-plate-forme en plus (voir le tableau de compatibilité des processeurs). Il a en effet été porté sous processeur Intel (du 386 au Xeon, en passant par les Pentium de première génération), mais aussi PowerPC, Alpha, Sparc, ARM, MIPS, etc. Ce système d'exploitation est multi-tâche

et multi-utilisateur et sa robustesse lui permet d'être utilisé pour les applications critiques. La disponibilité du code source est aussi très utile, notamment pour effectuer des rectifications en cas de bogue ou de défaut de sécurité. C'est la raison pour laquelle les armées américaine et française utilisent Linux.

Tableau 2. La robustesse de Linux démontrée par une étude anglaise[16]

Environnement	*Fiabilité*
Win 95 + Professional Office	Une défaillance toutes les 42 mn ; redémarrage nécessaire dans 28% des cas
MacOS + Ms Office	Une défaillance toutes les 188 mn ; redémarrage dans 56% des cas
Unix (diverses plates-formes)	Moins d'une défaillance par an ; pas de redémarrage nécessaire
Linux	Aucun incident relevé en 3 mois

Les applications les plus courantes

Linux peut constituer une alternative ou un complément solide à NT ou Novell Netware. Unix libre et quasi gratuit, Linux est donc devenu le serveur de choix pour ceux qui n'ont pas les moyens d'un Unix propriétaire mais souhaitent bénéficier d'un niveau de fiabilité au moins égal à celui des meilleurs Unix. Des solutions bâties autour de Linux seront présentées en détail au chapitre 5.

En réseau d'entreprise, Linux peut servir de :

- **Serveur d'applications**. Linux a hérité de la puissance d'Unix. Or, depuis toujours, Unix est un OS multi-utilisateurs, ce qui veut dire que l'on peut se connecter à plusieurs sur un même serveur et que chaque utilisateur dispose d'un espace propre pour faire tourner ses programmes. Cette fonctionnalité, qui a d'ailleurs permis à Unix de remplacer en partie les *mainframe*, est absente des versions standard de Windows NT. Elle permet à des utilisateurs dotés de terminaux X ou d'ordinateurs de réseau (*Network Compu-*

16 Source : « Software failures. Follies and fallacies ». Les Hatton, IEE Review, Mars 97

ter) d'accéder simultanément à des applications graphiques hébergées sur un serveur central commun.

- **Serveur de bases de données**. Comme Unix, Linux est un OS stable. On peut donc gérer des applications critiques sur des serveurs de bases de données conformes à la norme SQL qui sont parfois des logiciels libres (PostgreSQL), des logiciels semi-libres (MySQL), des *freeware* (Solid, PrimeBase) ou des logiciels propriétaires (Adabas, Pick, Oracle, etc.).
- **Serveur de fichiers**. Linux peut servir à enregistrer un fichier à travers le réseau sur le disque dur du serveur ou à accéder à une batterie de CD-ROM gérés par le serveur. Comme les Unix propriétaires, Linux supporte de nombreux protocoles de communication, ce qui permet de travailler en environnements hétérogènes : SMB pour Windows et IBM OS/2, IPX/SPX pour Novell Netware, Appletalk,/AppleShare pour l'environnement Apple, NFS pour les autres postes Unix. Le noyau de Linux supportant les modes de stockage redondant Raid 1, Raid 4 et Raid 5, Linux est adapté à des applications critiques.
- **Serveur d'impression**. Linux gère la plupart des protocoles d'impression et peut, dans un réseau hétérogène, gérer les files d'impression pour les différents utilisateurs d'une imprimante.

Linux est également un excellent système d'exploitation pour la gestion d'un réseau de type Intranet et pour des applications de *groupware*. Sa très large utilisation par les sociétés de services liées à Internet en font l'un des logiciels les plus testés et les plus sécurisés. Il peut servir notamment de :

- **Routeur**. Une disquette sur laquelle est enregistrée une version compacte de Linux permet de transformer n'importe quel PC en un routeur performant et modulaire capable de gérer les protocoles de communication les plus connus, dont TCP/IP, à travers

des cartes réseau ou des modems. La fonctionnalité de *masquerading* intégrée à Linux permet même de partager en toute sécurité et de façon transparente un accès individuel à Internet sur le réseau local d'une école, d'une entreprise ou d'une collectivité locale.

- ***Firewall*** (pare-feu). Les composants de communication de Linux comportent un *firewall* (pare-feu) intégré, des systèmes de filtrage et des technologies de protection. Ils ont la réputation d'être parmi les mieux testés, les plus solides et les moins vulnérables ce qui conduit à faire de Linux l'un des meilleurs *firewall* du marché. Par ailleurs, divers composant de chiffrement (ssh, ssh2, IPSEC, etc.) permettent de sécuriser de façon totalement transparente une communication sur Internet.
- **Serveur d'Intranet**. La plupart des outils de communication d'Internet ont été développés sous forme de logiciels libres pour Unix. Ils sont naturellement disponibles pour Linux : serveur de nom de domaine (DNS), serveur d'adresses IP (DHCP), serveur de courrier électronique, listes de diffusion, fax en réseau, annuaire (LDAP), serveur Web etc. De nombreux langages de script permettent de combiner ces serveurs à une base de données pour déployer des applications sophistiquées de *groupware* et de *workflow*.

Linux est aussi utilisé pour bâtir des stations de travail d'autant plus performantes que le noyau de Linux gère les ordinateurs multiprocesseurs.

- **Station bureautique**. De nombreux logiciels libres (Maxwell, SIAG, KOffice, etc.) ou propriétaires (Star Office, Applix, etc.) de bureautique offrent un confort de travail équivalent à n'importe quelle suite bureautique sur un système d'exploitation propriétaire avec, en prime, le choix de l'environnement de travail

et de l'interface utilisateur. Par ailleurs, divers émulateurs ainsi que des logiciels d'accès distant permettent d'utiliser sous Linux des applications développées pour MacOS, Windows ou d'autres OS propriétaires.

- **Station de développement.** Linux est le plus souvent fourni avec un environnement de développement complet, des compilateurs pour la plupart des langages (C, C++, Fortran, Java, Smalltalk, Cobol, etc.) et des outils graphiques de développement. D'excellentes bibliothèques graphiques permettent de développer sous Linux des applications tournant à la fois sous Windows, MacOS et tous les Unix. Des jeux vidéo célèbres (Doom, Quake, Abuse, etc.) ont d'ailleurs été développés sous Linux avant d'être diffusés pour Windows et MacOS.
- **Station scientifique**. La plupart des applications scientifiques ou d'ingénierie ont été développées pour Unix. Elles peuvent donc tourner sous Linux où elles bénéficient d'un environnement stable et d'un matériel peu onéreux. Par ailleurs, la plupart des grandes applications scientifiques (Scilab, Matlab, Mupad, Maple, Mathematica, etc.) sont disponibles sous Linux.
- **Station multimédia**. Le créateur de son, de musique ou d'images et d'animations 3D trouvera sous Linux des outils particulièrement performants et le plus souvent gratuits.

Enfin, en mettant une version compacte du système d'exploitation sur une disquette, on peut transformer un vieux PC 386 ou PC 486 en terminal X ou en *Network Computer*. Linux permet alors d'accéder depuis une machine peu puissante à des applications modernes telles que Star Office ou Netscape qui sont, elles, hébergées sur un serveur d'application sous Linux ou sous tout autre Unix.

Bien choisir sa distribution Linux

Le foisonnement des logiciels libres et l'accès au code source a donné naissance à la notion de distribution pour faciliter l'installation de Linux et éviter de devoir télécharger plusieurs gigaoctets de données qu'il faut ensuite compiler. Une distribution consiste à placer sur quelques CD-ROM une collection de logiciels libres, et parfois de logiciels commerciaux, et de fournir une disquette capable d'effectuer automatiquement l'installation de ces logiciels pour aboutir à un environnement de travail directement utilisable.

La plupart des distributions[17] sont vendues entre 200F et 500F sur des CD-ROM (voir Annexe 2 pour une liste de points de vente) mais sont aussi téléchargeables gratuitement sur Internet ce qui peut être utile pour effectuer une mise à jour, à condition de disposer d'une liaison rapide.

SuSE : la plus complète

SuSE est une distribution européenne multilingue (français, anglais, allemand, espagnol) lancée en 1992. Elle est livrée avec une disquette d'installation rapide bien conçue, capable d'effectuer une installation aussi bien à partir du jeu de CD-ROM (5 CD-ROM pour €50) qu'à partir d'un serveur de fichier en réseau ce qui peut être utile lorsque l'on souhaite installer Linux sur un PC sans CD-ROM. L'installation se fait le plus souvent en mode *plug and play*. Nous avons beaucoup apprécié la fonction d'analyse du matériel qui permet de découvrir les caractéristiques techniques détaillées d'un PC et de parvenir à installer Linux plus facilement sur un vieux PC dont les composants ne sont pas « *plug and play* ». Il existe aussi un CD-ROM de démonstration qui permet de démarrer SuSE Linux sans devoir l'installer sur un disque dur ou de bénéficier de toute la puissance de SuSE Linux en occupant seulement 50 méga-octets sur le disque dur du PC.

17 Sources comparatives : Infoworld, 8 Juin 1998 et linux-france.com

SuSE combine le meilleur des logiciels libres et des logiciels commerciaux pour Linux. Les principaux logiciels libres y sont livrés. L'utilisateur a le choix entre plus de 10 environnements graphiques différents dont KDE et GNOME. Le serveur Apache est désormais livré avec des modules d'optimisation (Perl, PHP) et de chiffrement (SSL 128 bits). Un serveur de fax en réseau est intégré par défaut ainsi qu'une application Java permettant de consulter les fax arrivés depuis n'importe quel poste.

Dans le domaine des bases de données et des progiciels d'entreprise, SuSE est probablement la distribution la plus riche puisque pas moins de 5 produits sont fournis : Adabas (propriétaire), Postgres (libre), MySQL (semi-libre), Yard, Solid. Ces serveurs sont complétés par une dizaine d'environnements graphiques de développement rapide dont HP Eloquence. SuSE est par ailleurs revendeur agréé des produits de Sybase, Informix et Oracle.

Des logiciels commerciaux de bureautique sont également proposés dont Star Office, Applixware, Wingz ainsi que des logiciels de création graphique (Gimp), d'animation 3D (Blender), d'édition sonore (SLab), etc. Logiciels libres et logiciels commerciaux sont clairement distingués par le logiciel de configuration, YaST, qui permet par ailleurs de configurer simplement tous les paramètres systèmes fondamentaux, de définir des installations type (serveur en réseau, *firewall*, station de travail, configuration minimale etc.) et d'ajouter ou de supprimer des modules après l'installation.

SuSE rémunère 4 développeurs pour assurer la disponibilité des pilotes pour les cartes graphiques les plus récentes du marché. SuSE est distribuée en Europe et aux Etats-Unis. Le réseau allemand de SuSE compte par exemple 3000 points de vente. En France, l'assistance téléphonique est assurée par la société Arkane Media[18] qui distribue aussi des configurations préinstallées (serveurs, bureautique, graphique, etc.). SuSE est

18 http://www.arkane-media.net

également disponible par correspondance auprès de MCD2[19].

Red Hat : la plus connue

La reconnaissance de Linux aux Etats-Unis, puis dans le reste du monde provient en grande partie de l'effort de marketing et du succès commercial de la société Red Hat outre-Atlantique depuis son lancement en 1994. La distribution conçue par Red Hat ne fournit normalement que des logiciels libres. L'acheteur peut donc en faire un produit commercial et le distribuer, voire le vendre, sans payer de royalties, mais à condition de fournir le code source des adaptations effectuées. Par exemple, Lectra, entreprise bordelaise, intègre la distribution Red Hat à ses stations de CAO textile.

L'environnement graphique KDE n'y a pas été intégré parce que que la bibliothèque Qt, nécessaire à son fonctionnement, n'était pas jugée assez libre alors que l'environnement GNOME l'était. Un changement des conditions d'usage de Qt intervenu en novembre 1998 devrait normalement conduire Red Hat à intégrer à terme l'environnement KDE[20] aux côtés de GNOME. En attendant, des versions dérivées de la distribution Red Hat comme Mandrake (voir plus bas) offrent en standard le confort de l'environnement KDE.

L'une des spécificités de Red Hat est le logiciel d'installation des paquetages logiciels, le *Red Hat Package Manager* (RPM), qui facilite considérablement la diffusion, l'installation et la suppression de composants logiciels sous Linux. La technologie RPM a d'ailleurs été reprise par la plupart des autres distributions à l'exception de Debian[21]. Le jeu de CD-ROM standard ($49 pour 3 CD) comprend d'une part la distribution Red Hat (2 CD) qui ne contient que des logiciels libres (Apache, sendmail, Perl, Gimp etc.) et d'autre part un CD d'applications propriétaires en

19 http://www.mcd2-diff.fr

20 Sous la pression de ses utilisateurs allemands, Red Hat a intégré KDE à sa distribution allemande.

21 La distribution Debian utilise par défaut un système de paquetage qui lui est propre tout en restant largement compatible avec les paquetages RPM.

version de démonstration ou limitée (Applix, Sybase Adaptive Server, Wordperfect, Communigate etc.).

La distribution Red Hat est très appréciée pour son support technique aux Etats-Unis. L'un des développeurs de Red Hat est payé à temps complet pour participer à l'évolution du noyau de Linux. La société a par ailleurs la réputation de bien intégrer les modifications demandées ou apportées par les utilisateurs.

Debian : la plus libre

Debian est la seule distribution Linux majeure conçue de façon coopérative, par une communauté de 400 développeurs. Des règles d'ingénierie distribuée très précises assurent une grande cohérence et un excellent contrôle qualité à cette distribution. Debian sort régulièrement gagnante dans les comparaisons techniques mais la faiblesse de son mode de diffusion commerciale la rend difficile d'accès. Debian est probablement mieux adaptée aux professionnels (SSII, constructeurs, électronique embarquée) qu'aux particuliers. Linux France[22] note que « d'importants intervalles de temps séparent cependant parfois deux mises à jours stables et sa prise en main est parfois délicate, surtout pour les débutants. Mais une fois passés les premiers obstacles, on ne veut paraît-il plus en changer. »

Debian fournit une bonne base pour confectionner sa propre distribution généraliste ou spécialisée. MNIS propose ainsi une distribution fondée sur Debian fournie avec une disquette d'installation simplifiée et un environnement francisé autour de l'environnement graphique KDE. Debian sert également de base à la distribution Pingoo utilisée par le projet Linux-Edu[23] d'implantation de Linux dans les établissements scolaires en Haute-Savoie (voir plus bas).

---

22 http://www.linux-france.com/article/choix-distri/
23 http://www.linuxedu.org

Turbo Linux : la facilité avant tout

Développée par la société japonaise Pacific Hi-Tech, Turbo Linux est la distribution préférée des Japonais et serait la deuxième distribution la plus vendue au monde derrière Red Hat. Oracle l'a donc sélectionnée, aux côtés de Red Hat, SuSE et Caldera, pour le portage de ses applications.

Turbo Linux est à la fois disponible pour les PC à processeur compatible Intel et pour les Macintosh à processeur PowerPC. Il existe des versions en japonais et en anglais. Les versions nippones contiennent divers logiciels de saisie des caractère Kanji réalisés par des grands groupes japonais (Omron, Kyocera, Ryobi, etc.). La suite bureautique Applixware, livrée avec la version Pro de Turbo Linux, gère parfaitement les langues asiatiques.

La principale particularité technique de Turbo Linux est l'extrême simplicité de son installation. Sans rien comprendre aux caractères japonais, nous sommes parvenus à installer Turbo Linux en dix minutes sur un PC en cliquant « OK » à chaque question. Tout a été configuré automatiquement y compris la carte graphique. A titre de comparaison anecdotique, les possesseurs du PC sur lequel nous avions installé Linux ont mis malheureusement plus de 6 heures pour réinstaller Windows 95 en français.

Caldera Open Linux : la plus commerciale

La société Caldera édite une distribution Linux combinant logiciels libres et commerciaux comprenant notamment KDE, Star Office et Sybase Adaptive Server Enterprise. La particularité de Caldera est de se mettre en avant comme une société commerciale capable de proposer un service équivalent à celui de grands éditeurs de logiciels propriétaires. Divers contrats de services et de maintenance sont donc proposés systématiquement. Dans le même esprit, Caldera met en avant la compatibilité de son offre avec la norme iBCS qui permet d'utiliser sous Linux des ap-

plications développées pour un Unix propriétaire (ex : SCO).

Caldera propose par ailleurs une offre intégrée de serveur Linux compatible Novell Netware. Ce produit, largement propriétaire, supporte parfaitement le protocole IPX, permet une configuration des clients Netware et incorpore des outils d'administration de serveur pour réseau Novell. L'offre Caldera nous semble donc bien adaptée aux migrations d'un environnement propriétaire Novell vers un environnement libre Linux.

LinuxPPC : Red Hat Linux pour le Mac

Red Hat Linux n'est pas directement disponible pour les ordinateurs à processeur PowerPC. Aussi, le développeur Paul Mackerras a-t-il porté les principaux composants logiciels de Red Hat Linux pour les ordinateurs à processeur PowerPC tels que les Macintosh. On y trouve notamment KDE, Netscape Communicator, Gimp, Apache, etc.

La version 5 de LinuxPPC devrait pouvoir démarrer directement à partir du CD-ROM d'installation. Il n'est donc plus nécessaire d'installer Linux sur un disque dur et de le configurer pour le tester ce qui constitue un grand progrès, y compris par rapport au CD-ROM de démonstration de SuSE qui nécessite tout de même une configuration de la carte graphique. Le logiciel BootX est désormais préconisé pour les Macintosh. Il permet de lancer LinuxPPC aussi simplement qu'une application MacOS ou de choisir au démarrage le système d'exploitation que l'on souhaite utiliser.

LinuxPPC offre des performances extraordinaires sur les Power-Macintosh G3 dont il reconnaît les cartes graphiques accélérées. Il fonctionne aussi désormais sur le iMac dont il reconnaît l'interface USB. Les Macintosh anciens lui préféreront MkLinux, la version de Linux développée par Apple autour d'un micro-noyau MACH et compatible avec les premier Power Macintosh (voir chapitre 4).

Mandrake : la Française

Conçue par Gaël Duval, il s'agit d'une méta-distribution intégrant Red Hat, l'environnement graphique KDE, des utilitaires facilitant l'accès aux disquettes et aux CD-ROM et un programme de configuration. La version 5.3 (baptisée « Festen ») comprend aussi des outils libres pour le SOHO : WordPerefect, l'éditeur WYSIWYG Klyx, basé sur LaTeX, le traitement de texte Maxwell, la suite bureautique SIAG, l'émulateur Windows WINE, Sane (accès aux *scanners*), l'agenda personnel Korganizer, Ksendfax (téléfax) et Kvoice (communication vocale via modem). Selon son auteur, « elle se configure automatiquement en français et gère sans problème le clavier Azerty ». Nous avons particulièrement apprécié la bonne conception des menus, la francisation et les applications livrées en standard qui permettent de disposer immédiatement d'un environnement de travail compact et utilisable. Mandrake est la première distribution au monde à proposer la version 2.2 de Linux et la version 1.1 de KDE.

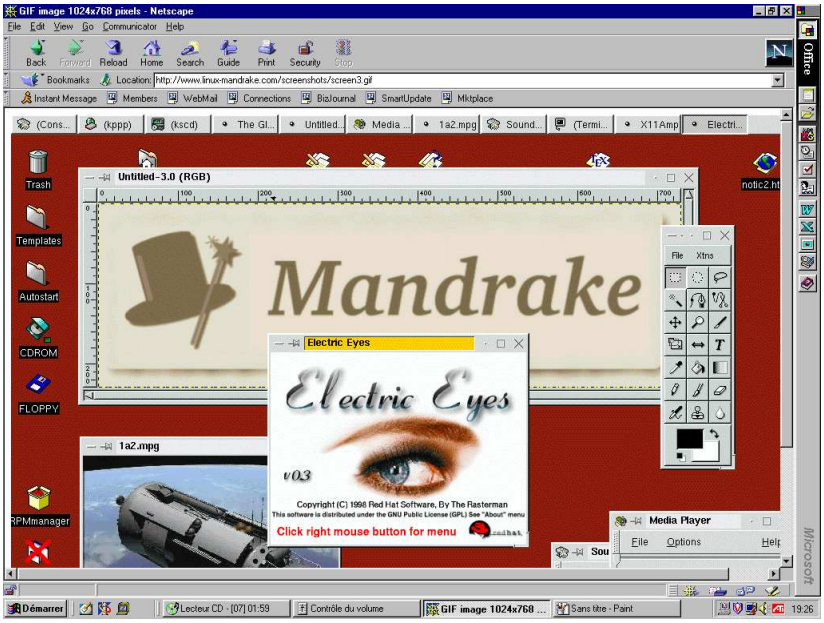

Figure 4. Linux Mandrake

Tableau 3. Compatibilité des processeurs et des distributions — les portages en cours ou disponibles indirectement sont entre parenthèses

	x86	*Alpha*	*ARM*	*68K*	*MIPS*	*PowerPC*	*Sparc*	*UltraSparc*	*SH*[24]
Debian	X	X	(X)	X	(X)	(X)	X	(X)	
SuSE	X	(X)	(X)[25]						
Red Hat	X	X	(X)	(X)	(X)	(X)	X	(X)	
LinuxPPC						X			
MkLinux	X					X			
Turbo Linux	X					X			
Caldera	X								

La guerres des mondes n'aura pas lieu

Avec Linux, la guerre des religions n'aura pas lieu car Linux fonctionne sur tous les matériels, communique avec la plupart des systèmes d'exploitation et émule presque tout.

Grâce au protocole Samba **(**SMB**)**, on peut gérer des postes clients Windows à partir d'un serveur sous Linux qui se comporte alors comme un serveur NT. Silicon Graphics a d'ailleurs adopté Samba dans son offre standard. Linux peut aussi émuler un serveur Novell, comme le prouve l'offre de Caldera ou l'émulateur MARS. Enfin, Linux permet de réaliser un serveur AppleShare très performant avec le kit Netatalk. Apple publie d'ailleurs sur son propre site un article[26] encourageant à utiliser des PowerMacintosh G3 sous Linux comme serveur de fichiers haute performance.

On constate la même ouverture au niveau des applications. Linux étant conforme à la norme Posix, les solutions sous Unix peuvent être portées sous l'OS libre. La compatibilité binaire avec SCO, mise en avant par Caldera et disponible sur les autres distributions, permet de faire tourner les applications conçues initialement pour cette plate-forme. Cer-

24 Hitachi annonce que Linux tourne sur ses processeurs SH (SH3, SH4) mais aucune grande distribution ne les prend en compte.
25 http://www.msnbc.com/news/224931.asp
26 http://www.apple.com/publishing/internet/kai/

VRAI OU FAUX : Linux fonctionne très bien sur un PC 386

C'est exact si on l'utilise comme un moyen de reconvertir son vieux 386 en Terminal X (voir glossaire) ou en serveur Web, c'est-à-dire pour un usage équivalent à celui d'un petit serveur sous Windows NT. Pour un poste bureautique autonome simple (suite bureautique SIAG ou Maxwell), un 486 est suffisant à condition d'utiliser un environnement graphique austère (twm, plan, etc.). Pour un poste bureautique autonome sophistiqué (Star Office, Netscape Communicator, KDE, etc.), un Pentium est nécessaire car toutes ces applications sont très gourmandes en puissance. La configuration minimale d'un PC multimédia « Lintel » est plutôt de 32 Mo, soit autant que pour Windows ou MacOS.

tains logiciels d'Adobe, dont FrameMaker, peuvent d'ailleurs tourner sous Linux alors qu'ils sont proposés pour SCO. Linux supporte aussi Java. Plusieurs machines virtuelles sont d'ailleurs disponibles et des éditeurs réputés comme Metrowerks ont offert à la communauté Linux leur technologie d'accélération Java.

Autre solution viable sous Linux : l'émulation. DOSEmu permet d'utiliser des applications DOS et Windows 3.1. Les applications conçues pour Windows 95 ou Windows NT sont utilisables soit à travers l'émulateur WINE (encore instable), soit à travers un client d'accès à un serveur d'application (Citrix, jBridge, VNC etc.). On peut même utiliser des applications Macintosh sous Linux grâce à Executor ou SheepShaver sans parler des myriades d'émulateurs de consoles de jeu vidéo (GameBoy, jeux d'arcade, etc.) et de micro-ordinateurs familiaux (ZX81, Commodore 64 etc,)

Enfin, pour un usage personnel, rien ne vaut le *dual boot* (double amorçage) qui permet de partager un même disque dur entre Windows et Linux ou entre MacOS et Linux et bénéficier avec une même machine du meilleur d'un environnement libre et d'un environnement propriétaire.

Bureautique sous Linux

Longtemps, aucune application bureautique n'a été portée sur Linux. Comme les applications bureautiques n'intéressaient pas grand monde dans le milieu de la recherche, aucun logiciel libre n'était disponible. L'une des premières exceptions propriétaires est le traitement de texte Corel Wordperfect, *challenger* malheureux de Word, et disponible depuis longtemps en version Linux. Mais il ne s'agissait que d'un traitement de texte et non d'une suite complète avec tableur, logiciel de présentation etc.

De nombreux outils commerciaux

Aujourd'hui, on trouve de nombreuses suites bureautiques propriétaires très complètes pour Linux. Les plus connues sont Applixware et Star Office.

Applixware comprend un tableur, un traitement de texte, un logiciel de présentation, un éditeur HTML et des fonctions bases de données. Il offre la possibilité d'importer des documents au format Microsoft Office, y compris Office 97.

Star Office comprend lui aussi un traitement, un tableur, des logiciels de présentation, d'illustration, de mise en page et de traitement d'image. Il propose aussi des fonctions de base de données relationnelle, de gestion de mail et de navigation sur les forums de discussions et sur le Web, ainsi qu'un gestionnaire graphique de fichiers similaire à l'explorateur de fichiers de Windows. Disponible en français, cette suite n'a rien à envier à sa concurrente Microsoft Office. La version 5.0 offre qui plus est des effets 3D qu'on ne trouve pas sur son équivalent Microsoft.

Derrière Star Office se trouve une société allemande, Star Division. Au départ, il s'agissait surtout d'une initiative germano-allemande, le slogan de Star Division étant « *software auf gut Deutsch* » (« du logiciel en bon

Polytechnique adopte Applix

A Polytechnique, dans la salle des stations d'enseignement, c'est sur la suite bureautique Applixware que la future élite française saisit ses rapports. L'Ecole militaire a en effet opté pour Linux pour une partie de son parc de stations de travail et cherchait à les enrichir de suites bureautiques. Cet été, Erian Concept, la société de services du groupe Les Logiciels du Soleil (qui édite la version française de Red Hat), a donc déployé la solution d'Applix sur 25 machines.

Pour les quelques étudiants déjà familiarisés à Office, « l'interface a paru un plus frustre, mais ils ont mis peu de temps à apprendre à l'utiliser », indique l'intégrateur. Restait toutefois à résoudre le transfert des fichiers personnels entreposés sur des serveurs sous Unix commerciaux. « L'intégration d'Applixware dans notre réseau Unix (NFS) s'est faite correctement», indique Gérard Guillerm, responsable des machines d'enseignement à l'X. Nous n'avons pas à regretter ce choix. Les outils sont largement utilisés en libre service ».

Interview par email réalisée à l'occasion d'un article sur la bureautique sous Linux. Voir « Des suites bureautiques pour Linux ». Le Monde Informatique du 5 février 1999.

allemand »). Star Division avait d'abord conçu une suite bureautique pour OS/2, Unix et Windows destinée aux grands comptes, essentiellement des banques. C'est donc tout naturellement, que Star Division a aussi effectué un portage sous Linux, qui ne nécessita pratiquement aucun effort de redéveloppement.

Star Division a alors utilisé deux jokers pour concurrencer Microsoft : faire tourner Star Office sur tout, y compris Linux, et le donner aux étudiants ou aux particuliers pour qu'ils y goûtent et continuent de l'utiliser une fois arrivé dans l'entreprise. Depuis, Star Division a conclu avec Sun Microsystems un accord de partenariat. Star Office est donc sur le point de devenir l'un des seuls concurrents sérieux à Microsoft Office.

Pour le reste, Applixware comme Star Office sont des produits propriétaires s sans leur code source ce qui rend l'utilisateur dépend du bon vouloir de l'éditeur pour le portage sur d'autres processeurs que les processeurs de type Intel. Applix a dans ce domaine une longueur d'avance

puisque ses produits sont aussi disponibles pour les versions de Linux pour processeurs Alpha, PowerPC et SPARC.

L'essor des interfaces graphiques

Le phénomène qui déclenchera probablement la montée en puissance d'une offre de suites bureautiques sous Linux est l'essor récent des interfaces graphiques visant à offrir sous Linux un confort de travail équivalent à celui d'un système propriétaire comme MacOS ou Windows. Deux projets de logiciels libres sont actuellement en compétition : KDE et GNOME.

KDE (*Kool Dekstop Environement*) est un projet particulièrement abouti puisque c'est le seul à proposer immédiatement une panoplie d'outils graphiques aussi complète que celle que l'on trouve en standard sur un système propriétaire. KDE a été lancé en octobre 1996 en Europe par des étudiants allemands et scandinaves. Le choix technologique de la bibliothèque Qt leur a permis de développer en un temps record un environnement complet et de prendre en compte les contraintes internationales complexes des langues européennes et asiatiques. Qt étant par ailleurs disponible sous Windows et fourni avec de nombreux outils de développement, les logiciels libres développées pour KDE sont portables sous Windows par recompilation du code source.

Le projet KDE a rencontré un succès fulgurant. Plus de 200 applications ont été développées dont des logiciels de courrier électronique, de traitement de texte, des jeux, etc. et tous peuvent interagir par glisser-déposer (*drag and drop*). KDE est disponible en plus de 25 langues. La version 1.1 de KDE intègre des technologies avancées d'interface utilisateur qui ne sont pas encore disponibles sous MacOS. KDE a été naturellement adopté comme interface utilisateur par défaut dans la plupart des distributions Linux a l'exception de notoire de Red Hat.

En effet, certains penseurs des logiciels libres estiment que le choix de

KDE ou GNOME : lequel adopter ?

Le choix entre KDE et GNOME n'en est plus un qu'en termes d'idéologie. Le fait que KDE, après Linux, soit d'origine européenne n'est probablement pas étranger aux discours « anti-KDE » par certains partisans du logiciel libre outre-Atlantique. Le fait que Qt soit diffusé sous une licence distincte de la licence GPL (voir Chapitre 5) continue à alimenter des discussions interminables sur la définition exacte d'un logiciel libre. Le point de vue de Linus Torvalds nous semble le plus raisonnable : tout en indiquant sa préférence pour KDE, l'inventeur de Linux respecte la liberté de développer un environnement concurrent à condition de ne pas pratiquer de terrorisme intellectuel.

D'un point de vue technique, les développeurs de KDE et de GNOME se sont entendus sur une norme commune pour le copier-coller. Une application développée pour KDE fonctionnera très bien sous GNOME et vice-versa. Les utilisateurs n'ont donc pas à s'inquiéter lorsqu'ils optent pour l'un ou l'autre des environnements. Quant aux développeurs, Qt leur permet de développer un logiciel fonctionnant à la fois sous Linux et Windows mais nécessite l'emploi du langage C++, Perl ou de Python alors que la technologie GTK peut être utilisée avec n'importe quel langage mais ne fonctionne pour l'instant que sous Unix.

La compétition entre KDE et GNOME relève donc d'une saine émulation entre deux technologies interopérables.

la bibliothèque Qt, qui n'était pas alors un logiciel libre, menaçait l'économie du logiciel libre. D'autre penseurs affirment au contraire que le choix de Qt ne posait pas de problème dans la mesure où il était relativement aisé d'en produire un clone et que l'éditeur norvégien de Qt s'engageait par contrat à ne pas tenter de s'approprier les développeurs de KDE.

Une contre-projet, GNOME, a alors été lancé à partir de la technologie GTK développée initialement pour le logiciel de retouche photographique Gimp. Ce projet a bien progressé au cours de sa première année mais connaît pour l'instant moins de succès que KDE à la fois en termes de fonctionnalités, de stabilité et d'adoption. En revanche, son existence et l'initiative lancées par quelques développeurs pour cloner Qt sous

forme de logiciel libre ont conduit Troll, l'éditeur de Qt, à diffuser sa bibliothèque sous forme de logiciel libre. Aussi, Red Hat serait sur le point d'intégrer KDE à son offre.

L'offre bureautique libre

L'arrivée d'environnements graphiques conviviaux a à la fois révélé des outils bureautiques libres restés longtemps inconnus et stimulé le développement de nouveaux outils.

Les principaux outils bureautiques libres sont le traitement de texte Maxwell, la suite bureautique SIAG, le logiciel de présentation MagicPoint. S'ils ne proposent pas toutes les fonctionnalités d'une suite bureautique propriétaire, ils n'en restent pas moins d'excellents outils de bureautique élémentaire ou d'enseignement de la bureautique. Leur faible besoin en puissance de calcul les destine tout particulièrement aux configurations modestes, comme celles à base de processeur 486.

Plus ambitieux, le projet KOffice vise à produire une suite bureautique avancée dans le cadre du projet KDE. La technologie qui a été choisie repose sur la norme CORBA, un format de document composite et un mécanisme de composants logiciels similaires à OLE (Microsoft) ou OpenDoc (Apple/IBM). La suite, encore au stade alpha ou beta, est cependant moins complète que Star Office.

Il est probable que la bureautique, tout comme les systèmes d'exploitation ou les navigateurs Web, rejoignent à terme la communauté des logiciels *Open Source* tant le besoin d'une suite bureautique performante est universel. L'essor de KOffice ou d'un autre logiciel libre comme standard du marché dépendra principalement de la capacité des éditeurs propriétaires à libérer suffisamment tôt le code source de leurs produits propriétaires pour conserver leur part de marché.

Tout pour le Web

Les stars de l'Internet sont des logiciels libres. Le serveur Web Apache, principal concurrent de Microsoft Internet Information Server et Netscape Enterprise Server, domine le marché avec environ 50% des sites Web gérés. Apache est disponible sur Linux, BSD mais aussi MacOS et Windows NT. Aux Etats-Unis, 26% des prestataires d'accès Internet utilisent Linux, selon une étude Infobeads. En Europe, 27% des serveurs Web sont sous Linux, selon une enquête de Keso. En France, Voila, le site portail de France Télécom, hébergé par la société Echo[27], est géré au moyen du moteur de recherche Extense[28] sur des serveurs Linux multiprocesseurs founis par le constructeur français Getek[29]. Quant à BSD, il est utilisé pour gérer le site Web le plus visité au monde, Yahoo!.

Apache

Apache permet de concevoir des serveurs de pages statiques mais aussi des pages dynamiques grâce à divers modules d'optimisation largement plus efficaces que la technologie CGI (Common Gateway Interface) habituellement utilisée par les serveurs Web. Les modules Perl, PHP ou Python d'Apache sont fréquemment employés pour la publication de bases de données sur le Web ou le développement d'applications de *groupware* et de *workflow*.

Perl est un langage de script développé par le programmeur Larry Wall à la fin des années 80 pour le traitement des chaînes de caractères au moyen d'outil de recherche et de remplacement très puissants. Perl se prête remarquablement bien au développement d'applications interactives pour le Web et à la manipulation de bases de donnés. Perl est disponible pour tous les systèmes de type Unix, pour Windows et MacOS.

27 http://www.echo.fr
28 http://www.extense.com
29 http://www.getek.fr/

PHP est un langage de syntaxe proche du C qui, à l'instar de la technologie ASP de Microsoft, permet d'intégrer dans une page HTML des éléments dynamiques calculés lors de la consultation et notamment des fonctions de consultation de base de données. PHP est disponible pour tous les systèmes de type Unix et pour Windows.

Python est autre langage qui combine la puissance des modèles objet comme Java avec la puissance de manipulation de chaînes de caractères de Perl. Il tourne sur tout type de plate-forme, de l'ordinateur de poche Psion au supercalculateur Cray. On peut notamment l'utiliser pour créer des systèmes de consultation et de publications de bases de données objet via le Web.

Apache peut être complété sur Linux par de nombreux outils libres pour construire un serveur Web très complet :

- Sendmail : logiciel de transmission du courrier électronique (libre sauf sous Windows).
- Procmail : filtrage et traitement automatique du courrier électronique.
- Sympa : gestionnaire de liste de diffusion multilingue (gère correctement les accents du français)
- Wu FTP : serveurs de fichiers FTP
- InterNetNews : serveur de groupes de discussion (*news*)
- OpenLDAP : serveur d'annuaires (LDAP)

Sur Linux, toutes ces applications peuvent être combinées au moyen de Perl, PHP, Python ou n'importe quel autre langage de programmation et gérées à distance grâce à divers outils graphiques d'administration système.

Des serveurs libres et commerciaux

Apache semble avoir montré l'exemple à suivre pour éditer un serveur

d'Intranet. Les suédois de Roxen commercialisent désormais sous forme de logiciel libre un serveur Web à hautes performances : Roxen Challenger. Roxen a gagné le prix du meilleur serveur au Comdex 1998. IBM, après avoir adopté Apache pour son offre de commerce électronique, diffuse aussi sous forme de logiciel libre un serveur de courrier électronique sécurisé.

Les logiciels libres semblent être devenus la norme du marché dans le domaine du Web.

Cas d'écoles

Linux, BSD et les logiciels libres sont des plates-formes idéales pour l'enseignement. Il sont d'ailleurs dominants à l'université et connaissent un succès croissant dans les écoles, collèges et lycées.

Il y a à cela plusieurs raisons pratiques :

- La liberté de copier gratuitement des logiciels adaptés à tous les micro-ordinateurs, y compris ceux aux performances modestes, garantit des prix particulièrement bas.
- La disponibilité du code source facilite l'apprentissage des fondements de l'informatique, atout très apprécié dans l'enseignement supérieur.
- La liberté d'adapter le code source permet aux chercheurs d'expérimenter et de valoriser plus rapidement leurs travaux.

« Grâce à Linux, les machines sont facilement protégées contre les modifications des stagiaires. Il est facile de restaurer automatiquement les machines pour le stage suivant et d'empêcher les stagiaires de toucher aux configurations », note Jacques Le Marois sur son site Linuxbusiness.com[30].

30 http://linuxbusiness.com

Exemples à l'étranger…

L'Ecole des Trois Soleils, dans la banlieue de Montréal, a recyclé ses vieux PC 486 en terminaux X sous Linux. Leur manque de mémoire (4 Mo) est pallié par le fait que les applications (Communicator et la suite bureautique Applixware) sont hébergées sur un PC serveur à Pentium 166 Mhz, doté d'un gros disque dur (5 Go) et de 64 Mo de RAM.

Le gouvernement mexicain a décidé de déployer ce type de solution à grande échelle : dans le cadre d'un programme d'informatisation sur cinq ans, Scholar Net[31], 140 000 écoles primaires et collèges seront équipées de serveurs pour Terminaux X sous Linux. Un système sous serveur NT avec Windows 98 et Microsoft Office sur les postes clients serait revenu à environ $100 (550 FF) par poste pour les licences de logiciels. Avec Linux, il suffit d'acquérir un seul CD-ROM à $50 (275 F) et de le copier en toute légalité sur tous les serveurs et tous les postes. Le fournisseur de Linux, Red Hat, a signé un contrat d'assistance avec le réseau Scholar Net pour l'aider à adapter son offre aux contraintes des enseignants mexicains.

…Et en France

Le serveur Ecole-Info, conçu par Michel Quercia (de l'Académie de Dijon), est un serveur Linux destiné à être installé dans une école pour fournir l'accès Internet, le partage de fichiers et le partage d'imprimantes sur un réseau local. L'objectif est d'obtenir à terme un serveur dont l'installation et la mise en œuvre sont réalisables par toute personne, même si elle n'est pas spécialiste de Linux.

Linux-Edu[32] est une initiative qui consiste à déployer des serveurs sous Linux dans les collèges et lycées. Déployés d'abord en Haute-Savoie, ces serveurs assurent des fonctions de type Web, proxy, DNS,

31 http://www.linux.org.mx/arturo/scholar/
32 http://www.linuxedu.org

courrier électronique, de façon totalement transparente pour l'utilisateur. Il est d'ailleurs possible d'en assurer la maintenance à distance. Linux-Edu repose sur une distribution Linux, baptisée PingOO et basée sur Debian.

Programmes marchands

Un argument avancé en faveur des logiciels libres dans le système éducatif est de garantir l'indépendance entre le programme scolaire et les stratégies marchandes des éditeurs de logiciels propriétaires. En effet, le système scolaire est considéré par de nombreux éditeurs comme un moyen formidable d'habituer sa future clientèle à ses produits. Dans la mesure où le rôle de la formation informatique à école n'est probablement pas de conditionner de futurs consommateurs de logiciels mais d'ouvrir les esprits à la diversité des outils, l'usage de nombreux logiciels libres conçus selon des principes différents ne peut que contribuer à cet objectif.

L'offre de logiciels propriétaires contribue bien entendu à cette diversité. Aussi, l'usage exclusif des logiciels libres ne doit pas être imposé lorsque les budgets permettent l'acquisition de logiciels propriétaires. En revanche, il nous semble absolument indispensable d'installer systématiquement un système d'exploitation libre sur tous les micro-ordinateurs utilisés dans l'enseignement selon un principe de double amorçage. Il serait en effet dommage d'interdire aux écoliers, collégiens ou lycéens de pouvoir profiter de l'offre extraordinaire de logiciels libres, notamment dans le domaine scientifique, alors qu'un OS comme Linux n'induit aucun surcoût d'installation et contribue même à faire baisser les coûts d'exploitation d'OS propriétaires comme le montre l'expérience de Michel Quercia à Dijon.

Multimédia et création

La création de contenu multimédia, longtemps parent pauvre du logiciel libre, commence aussi à être gagnée par la vague libre.

Logiciels graphiques

Dans les laboratoires scientifiques, la nécessité de manipuler des photos issues de données satellite a donné naissance à de nombreux outils de traitement d'image. La recherche sur la synthèse d'images a elle fait naître de nombreux logiciels de rendu par *ray tracing* ou par *radiosité* de modèles tridimensionnels. Linux constitue une plate-forme de choix pour ces applications car il est assez facile de faire exécuter des taches sur plusieurs machines et/ou sur plusieurs processeurs.

La grande nouveauté est que ces applications libres, pas toujours très conviviales, ont évolué ou cédé la place à des applications dont la convivialité rivalise de plus en plus avec les standards propriétaires du marché et l'environnement Macintosh. L'exemple le plus connu le logiciel de retouches d'images Gimp. Il s'agit d'un clone libre de Photoshop qui a l'avantage d'être conçu autour d'une architecture plus modulaire grâce à laquelle le logiciel évolue très vite. On peut échanger de nombreux petits modules pour obtenir de nouveaux effets graphiques.

Dans le domaine de la 3D, PovRay faisait jusqu'à présent figure de référence et pouvait être piloté par des logiciels libres de modélisation comme Moonlight Creator. Blender, un *freeware* néerlandais pour Linux, change cette donne en offrant à tous les utilisateurs de Linux un environnement professionnel de modélisation, d'animation et de rendu. Le site Linux3D[33] référence les principaux logiciels de 3D sous Linux.

33 http://www.linux3d.org

Son et musique

Linux affirme sa force dans le domaine de la création sonore puisque de nombreux logiciels libres ou *freeware* de montage audionumérique sont disponible. The Sound Laboratory (*freeware*) et ses 64 pistes de *direct-to-disk* (enregistreur sur disque dur) rivalise avec les logiciels de haut de gamme utilisés dans les studios d'enregistrement sous MacOS. Multitrack, un équivalent libre, n'offre lui que 8 pistes. Des logiciels plus austères tels que Csound ou Cmix font référence dans le milieu de la recherche et offrent des fonctionnalités très intéressantes dans le domaine de la synthèse sonore. Ils évoluent progressivement par l'ajout d'outils graphiques et de fonctions de synthèse sonore en temps réel.

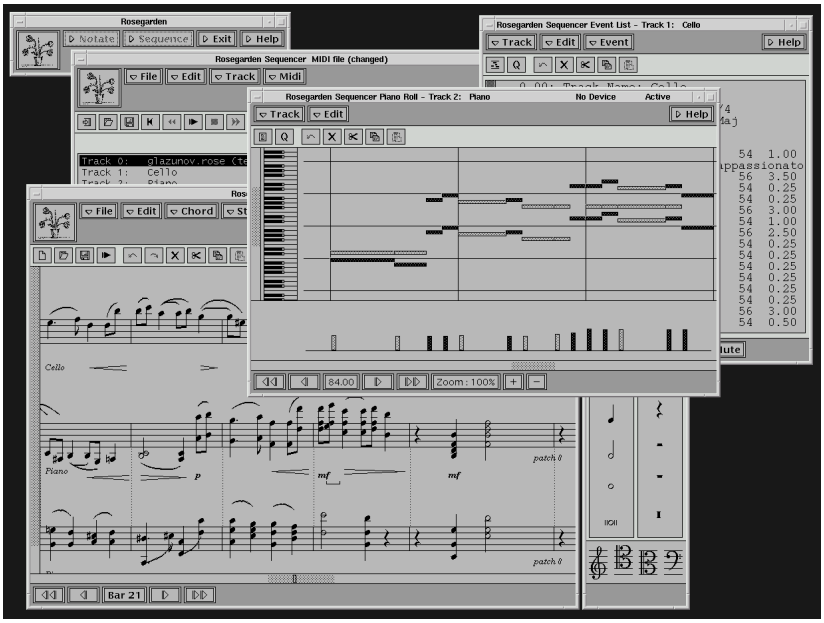

Figure 5. Composition musicale avec Rosegarden

Dans le domaine des séquenceurs MIDI et des éditeurs de partition, certaines versions du *shareware* Jazz sont disponibles avec leur code source alors que Rosegarden est un logiciel libre. KooBase est un projet

sur le point de proposer sous Linux une alternative viable au séquenceur propriétaire Cubase utilisés par de nombreux logiciels sous MacOS ou Windows. Divers logiciels de synthèse sonore peuvent compléter ces deux logiciels de référence pour fournir au compositeur un environnement très complet. Tout comme sous MacOS et Windows, la synthèse sonore en temps réel de sons analogique *vintage* sous Linux avec des logiciels de simulation de basses TB303, de boîtes à rythmes et même de synthèse par modèles physiques avec RTSynth

Hypermédia

Les sons et images ainsi créés sous Linux peuvent être assemblés sous forme de documents hypermédias, soit en utilisant la norme ouverte HTML et un éditeur adapté, soit en faisant appel à un logiciel propriétaire de création multimédia tel que Metacard. Le principal intérêt de la norme HTML réside dans la disponibilité de logiciels libres pour la rédaction et la consultation des documents hypermédias. La publication du moteur Gecko de Netscape en *open source* devrait d'ailleurs renforcer l'usage de cette norme pour la réalisation de présentations hypermédias multi-plate-formes. Mais pour ceux qui préfèrent une interface de type Hypercard, il vaut probablement mieux utiliser le logiciel propriétaire Metacard pour produire des applications similaires à celles que l'on peut produire avec Director ou Supercard avec en prime une véritable compatibilité multi-plate-forme (Unix, Linux, MacOS, Windows).

Bases de données

Les serveurs de bases de données les plus utilisés sur le Web sont des logiciels libres : PostgreSQL et MySQL.

PostgreSQL est un serveur de bases de données combinant un modèle relationnel et un modèle objet. PostgreSQL a été développé à partir

des travaux du professeur Michael Stonebraker de l'université de Berkeley sur les bases de données. Ces travaux avaient déjà engendré l'une des premières bases de données relationnelles : Ingres. Mais, le langage Quel utilisé par la version initiale de Postgres rendait difficile son adoption par un marché habitué à SQL. Deux étudiants, Andrew Yu et Jolly Chen, ont décidé de réécrire une partie de Postgres 4.2 pour le rendre compatible avec la norme SQL qui est le standard du marché. C'est ainsi qu'est né PostgreSQL en 1995. PostgreSQL offre des fonctions sophistiquées telles que les transactions, l'héritage entre classes, les procédure stockées. Ces fonctions ne sont généralement disponibles que sur des logiciels serveurs haut de gamme. Revers de la médaille : PostgreSQL est relativement lourd et coûteux en puissance de machine.

C'est pourquoi, de nombreux programmeurs lui préfèrent MySQL en raison de sa compacité, de sa rapidité et de sa légèreté. MySQL a été élue en 1998 meilleure base de données abordable par le webzine CNet[34]. Divers outils facilitant la publication de bases de données sont proposés dont certains permettent de manipuler les tables à partir d'un navigateur Web. MySQL est d'un point de vue juridique un logiciel libre (contrat de licence, code source ouvert) et d'un point de vue moral un *shareware*. Ses auteurs, la société suédoise TCX, demandent en effet aux utilisateurs qui en font un usage commercial de bien vouloir participer à son développement en achetant à TCX du service de conseil.

La fonction de base de données peut aussi être réalisée au moyen de langages de haut niveau comme Perl ou Python. De nombreux auteurs de site Web utilisent Perl pour gérer des fichiers simples tandis que d'autres auteurs affirment que Python, grâce à son mécanisme d'objets persistants, leur permet de développer bien plus rapidement une base de données publiée sur le Web que s'ils utilisaient un serveur relationnel comme MySQL ou PostgreSQL. Le serveur de *groupware* BSCW[35], entiè-

34 http://www.cnet.com
35 http://bscw.gmd.de

rement réalisé en Python, semble démontrer qu'il s'agit d'une alternative tout à fait sérieuse et portable.

PAO scientifique

La plupart des éditeurs de texte et des logiciels de PAO utilisés par le monde scientifique sont libres.

TeX est le plus connu d'entre eux. C'est un logiciel de typographie et un langage de programmation capable de calculer automatiquement toutes les règles de typographie pour obtenir une mise en page strictement conforme aux normes internationales des typogaphes. TeX gère donc la mise en page à la place de l'utilisateur.

LaTeX est une extension de TeX très utilisée pour la rédaction des articles scientifiques et des thèses de doctorat. LyX et KLyX sont des applications de type traitement de texte capables de produire au moyen d'une interface graphique similaire à celle de Word des documents aux formats LaTeX mais aussi aux formats HTML, XML et SGML pour la publication sur le Web. Notons que la réputation d'aridité de TeX ou de LaTeX est devenue infondée depuis qu'existent LyX et KLyX.

TeX et LaTeX permettent de produire de gros documents scientifiques mais aussi des livres. Par exemple, de nombreux manuels de l'éditeur O'Reilly sont mis en maquette avec TeX. Il suffit de définir le format d'une collection dans un style conçu avec TeX et tout livre publié dans ce cadre est mis en page automatiquement.

Plus généralement, TeX excelle dès que des mises en pages complexes doivent être appliquées automatiquement à un important corpus de texte bien structuré. Aucun traitement de texte du marché ne permet d'obtenir ce type de résultat aussi aisément et avec un résultat graphique aussi cohérent.

Programmation

La *Free Software Foundation* (FSF) de Richard Stallman est à l'origine de la plupart des outils de développement libres, utilisés d'abord pour les Unix commerciaux, ensuite pour Linux.

L'héritage de la FSF

Le plus connu est gcc, l'un des compilateurs C/C++ les plus utilisés au monde. Si les micro-ordinateurs sont toujours livrés sans compilateurs, les stations de travail Unix disposent normalement d'un compilateur C installé par défaut. Mais, au lieu de fournir un compilateur conforme à la norme ANSI, les constructeurs de stations de travail livraient à la fin des années 80 un vieux compilateur non-conforme et exigeaient des utilisateurs de payer pour bénéficier d'un compilateur récent. Au même moment, la FSF développait un environnement complet d'outils de développement. Le compilateur gcc de la FSF était compatible avec la norme ANSI, puis C++, Objective C etc. Comme il était d'une grande qualité et portable d'une station à une autre, il fut très rapidement adopté par la plupart des développeurs sur stations de travail.

On doit à la FSF de nombreux autres outils de développement qui ont aussi remplacé en tant que standards du marché leurs équivalents propriétaires issus des diverses lignées d'Unix. Les plus célèbres sont le gestionnaire de projets GNU Make (gestion de projets), le débogueur gdb, l'analyseur syntaxique Bison et l'éditeur de texte Emacs, etc.

Depuis, le développement de gcc a été pris en charge par une société commerciale, Cygnus, qui l'a rebaptisé egcs. Cygnus offre egcs sous licence GPL, mais vend des adaptations spécifiques (*custom development*) du produit à ses clients. Parallèlement, Cygnus fait évoluer egcs de manière à ce qu'il intègre la compilation de Java en code natif, afin que l'on puisse se passer de machine java virtuelle. Une étape d'autant plus aisée à franchir que la syntaxe de Java est déjà dérivée de celle de C++.

Peu à peu, les outils quelque peu spartiates de la FSF ont été complétés par des outils de développement à interfaces graphiques. Le débogueur DDD[36] permet par exemple de se promener visuellement dans toutes les structures de données de l'ordinateur, afin d'y déceler et d'y rectifier les dysfonctionnements d'un programme. DDD fonctionne avec les langages C, C++ et Fortran.

Tout pour le développement rapide

Les langages Perl et Python, déjà décrits dans ce chapitre, permettent de développer très rapidement des applications client/serveur ou des utilitaires de conversion de fichiers, surtout lorsqu'ils sont combinés à des bibliothèques graphiques telles que Qt ou GTK. A ces deux langages, il convient d'ajouter l'environnement Tcl/Tk qui, depuis bientôt dix ans, offre ce que Java promet sans le réaliser parfaitement : « écrire une fois, exécuter partout » (« *write once, run anywhere* »).

Les amateurs de développement en C++ et de *frameworks* ne sont pas en reste. La bibliothèque WxWindows permet de développer des applications qui pourront ensuite fonctionner sur n'importe quelle plateforme. La bibliothèque Qt permet elle de disposer d'une technologie de développement comparable à celle d'OpenStep mais disponible sur tous les systèmes d'exploitation à l'exception de MacOS. La plupart de ces outils de développement peuvent être complétés par des constructeurs interactifs d'interfaces graphiques et des environnements intégrés de développement, tous disponibles sous forme de logiciels libres.

Le seul domaine dans lequel l'offre de logiciel libre est, temporairement insuffisante, reste le développement d'applications Java mais l'annonce par Sun de diffuser Java sous forme de logiciel *Open Source* devrait conduire à stimuler les développeurs de logiciels libres à améliorer les performances de Java sous Linux et à proposer des environnements de

36 http://www.cs.tu-bs.de/softech/ddd/

développement rapides et efficaces sous licence libre. En attendant, d'excellents environnements de développement propriétaires, comme NetBeans ou VisaJ, sont disponibles et intégrés à certaines distributions Linux.

Tout pour l'image et le son

Le développement d'applications multimédias est grandement facilité par diverses bibliothèques libres ou semi-libres.

Pour le développement en 3D sous Linux, on peut utiliser une version propriétaire de la bibliothèque OpenGL, la norme standard du marché pour les applications 3D. Mais il existe aussi un clone libre de très bonne qualité, Mesa, disponible pour l'Unix libre mais aussi pour MacOS et Windows. Côté applications sonores, Open Sound System (OSS) est un excellent outil propriétaire mais il existe aussi des bibliothèques libres équivalentes comme ALSA.. Il existe par ailleurs de nombreuses applications libres dans le domaine de la 3D (rendu, modeleurs) et du son (synthèse, composition) qui peuvent servir de point de départ pour de nouveaux développements.

Sciences et ingénierie

Depuis plus de dix ans, Unix est le système de prédilection des chercheurs et des ingénieurs. Il n'est donc pas étonnant de retrouver sous Linux la plupart des grandes applications scientifiques ou techniques utilisées dans le recherche ou dans l'industrie.

Les grands logiciels de calcul scientifique (Scilab, Matlab, etc.) ou de calcul formel (MuPAD, Mapple, Mathematica, Macsyma, etc.) sont bien évidemment disponibles sous Linux. Notons que Scilab, un logiciel libre de l'INRIA, et MuPAD, un *freeware* allemand, sont aussi disponibles pour Windows et MacOS. Les logiciels libres ne s'adressent donc pas unique-

ment aux seuls utilisateurs de Linux. Il n'en reste pas moins que la plupart des logiciels libres scientifiques développés par les universités sont conçus pour le système Unix. L'essor de Linux les rend aujourd'hui accessibles au plus grand nombre.

Le succès croissant de Linux comme station de travail et les déboires rencontrés par certains utilisateurs avec Windows NT poussent aujourd'hui un nombre croissant d'éditeurs de logiciels techniques à porter leurs produits sous Linux. On trouve donc sous Linux des logiciels sophistiqués dans les domaines de la CAO (Arcad, Varicad, etc.), de la micro-électronique (Alliance VHDL, Green Mountain, etc.) , de l'architecture, des statistiques, etc. Une liste très complète de logiciels est publiée sur le site SAL[37].

BSD, l'autre Unix libre

BSD est une branche moins connue que Linux, mais tout aussi vivace, de la famille des freenix ou Unix libres[38]. Il s'agit d'ailleurs d'un descendant beaucoup plus direct d'Unix puisqu'il a été développé à partir de l'Unix de l'université de Berkeley qui était lui-même une variante de la version originale d'Unix cédée sous licence propriétaire aux universités américaines. Mais après une dizaine d'années, plusieurs thèses et plusieurs mémoires d'étudiants, il ne restait plus grand chose du code source originel d'AT&T, devenu depuis propriété d'Unix Systems Laboratories puis SCO. D'où l'idée d'exploiter commercialement ce nouvel Unix en l'adaptant aux processeurs Intel. Mais la première version viable de ce système d'exploitation, BSD/386, conçue en 1992 par la société Berkeley Software Design, fût retirée du marché à cause d'un procès intenté par Unix Systems Laboratories.

---

37 http://SAL.KachinaTech.COM/

38 Pour un bon historique des systèmes BSD, voir « Linux/etc", 1ère partie. Computer Bits. Volume 8. Numéro 9. Septembre 1998.

Qu'importe, l'idée de concevoir un Unix libre à partir de BSD était lancée. FreeBSD et NetBSD, puis OpenBSD, rejetons du précédent ne tardèrent pas à voir le jour ainsi que BSDi qui est leur équivalent propriétaire (voir figure ci-dessous).

BSD proche de Linux

La famille BSD a beaucoup en commun avec Linux. Elle utilise aussi le système de fenêtrage libre X-Window du consortium Xfree86 et les outils GNU comme le compilateur C egcs. Elle inclut d'ailleurs beaucoup de lignes de code présentes dans le noyau de Linux. Les systèmes BSD peuvent par ailleurs faire fonctionner des programmes Linux via une bibliothèque de compatibilité. Enfin, leur robustesse et leur disponibilité pour des processeurs anciens (386, 486, etc.) les prédisposent, comme Linux, à jouer les serveurs Intranet/Internet et/ou à recycler de vieux PC.

Tout comme Linux, les diverses variantes de BSD peuvent faire fonctionner des applications écrites pour les Unix propriétaires (SCO, Solaris, BSDi, OSF/1, SunOS, etc.). Le serveur Microsoft Front Page est par exemple utilisé par de nombreux hébergeurs sous BSD.

BSD is not Linux

Là où les deux rameaux divergent, c'est au niveau de la licence. Les BSD libres ne sont pas distribués sous GPL mais sous une licence qui leur est propre, la licence BSD. Moins contraignante, elle n'oblige pas à donner le code source des produits développés à partir de systèmes BSD, mais seulement à citer les noms de leurs auteurs. Cette particularité explique que la chaîne logicielle qui est issue de cette famille d'OS devient parfois propriétaire (voir chapitre 5).

Contrairement aux distributions Linux, les diverses versions de BSD ne partagent pas un noyau système commun. Chaque version de BSD (NetBSD, FreeBSD, OpenBSD) a son propre noyau et n'a pas de com-

patibilité binaire directe avec les autres. Il existe seulement un socle de code commun, issu de la version 4.4 BSD développée à l'université de Berkeley et plus ou moins incorporé aux différentes versions de BSD : 4.4 BSD-Lite.

Ces divergences entre versions s'expliquent aussi par un mode de développement pris en charge par une communauté restreinte, alors que la participation à l'évolution de Linux est *a priori* ouverte à tous.

Les différents BSD

NetBSD, la multi-plate-forme

Le cercle de développeurs de NetBSD a mis, dès le début, l'accent sur la portabilité. NetBSD est ainsi le seul Unix libre a supporter de longue date le port USB, utilisé par exemple par l'iMac ou les PC récents. La liste des machines supportées est impressionnante : Amiga, Atari, HP 9000/400, Macintosh avec Motorola 68k, DECstation 2100, 3100 et 5000, Sun3 ou Vax. Pour les processeurs, on compte notamment ARM, Alpha et, bien sûr, les Intel 386.

De cette souplesse, qui permet de reconvertir aisément de vieilles machines, découle aussi un handicap : NetBSD prend plus en compte l'adaptation aux matériels que les progrès dans les domaines logiciels.

FreeBSD, pour « booster » son PC

FreeBSD mise davantage sur la performance que sur la portabilité multi-plate-forme. Il était, d'ailleurs, à l'origine uniquement disponible pour 386. Depuis, ses auteurs ont choisi de le rendre aussi compatible avec les processeurs 64 bits Alpha (Digital-Compaq) et UltraSparc (Sun). « L'optimisation (de la puissance), combinée avec sa stabilité, ont donné à FreeBSD la réputation de pouvoir soutenir de très fortes charges. Si vous avez pour tâche de monter un serveur qui soit prêt à subir beaucoup d'attaque, FreeBSD est un excellent choix », écrit Terry Griffin

dans Computer Bits[39]. Rien d'étonnant à ce que Yahoo!, le site Web le plus visité au monde, soit géré sous FreeBSD.

FreeBSD joue plus la carte de la variété de l'offre logicielle que matérielle. L'offre commerciale de Walnut Creek (quatre CD-ROM) est livrée avec un nombre de logiciels qui n'a rien à envier à des distributions Linux comme SuSE. FreeBSD offre aussi des modules de noyau chargeables dynamiquement, ce qui permet de rajouter à chaud de nouveaux systèmes de fichiers, des protocoles réseaux ou des émulateurs binaires.

OpenBSD, le plus sécurisé

OpenBSD cherche à combiner les avantages de ses deux prédécesseurs, portabilité et stabilité, mais en mettant surtout l'accent sur la sécurité. Il reconnaît les mêmes plates-formes que NetBSD. Etant développé au Canada (comme, d'ailleurs NetBSD) autour de Theo de Raadt, il ne subit pas les restrictions d'exportation de technologies de chiffrement imposées aux Etats-Unis. Il peut donc constituer un bon choix pour un *firewall* ou un serveur sécurisé.

Quel BSD choisir?

Terry Griffin, consultant pour Axian Inc. a testé l'installation des trois BSD sur un PC avec Pentium 133, 64 Mo de RAM et 8 Go de disque dur. Pour les néophytes en matière d'Unix, FreeBSD est aussi aisé à installer que les distributions Linux les plus conviviales. A l'inverse, la configuration d'OpenBSD ou de NetBSD est particulièrement ardue et peu conviviale. Les utilisateurs les plus endurcis d'Unix y sont, paraît-il, tout à fait dans leur élément.

39 Voir « Linux/etc », 2ème partie. Volume 8. Numéro 10. Octobre 1998.

eCos, l'embarqué pour tous

Développé par la société Cygnus, eCos (Embedded Cygnus Operating System) est un système d'exploitation temps réel destiné à l'informatique embarquée, distribué sous licence *Open Source*. Il est adapté au contrôle des automobiles des appareils électroménagers, des magnétoscopes, des télévisions, des routeurs, des téléphones, des machines-outils industrielles etc. eCos est compatible avec la norme TRON, le standard ouvert le plus utilisé dans les produits électroniques de l'industrie japonaise.

Les utilisateurs potentiels d'eCos sont donc très nombreux mais resteront certainement peu connus du grand public. En effet, l'électronique embarquée envahit chaque jour un peu plus les objets quotidiens sans que nous nous en apercevions. Des analystes affirment que sur les 6 milliards de micro-processeurs en service dans le monde, moins de 600 millions se trouvent dans des micro-ordinateurs. Le marché potentiel d'eCos se trouve justement dans les 5 milliards 400 millions restants, un marché colossal mais peu médiatisé.

eCos peut être utilisé gratuitement par n'importe quel industriel ayant besoin d'un système d'exploitation compact. eCos est capable de gérer des contraintes de temps réel et intègre toutes les fonctionnalités de base pour accéder aux services Internet. Mieux, une technologie de modules paramétrables et d'interfaces interchangeables permet à eCos d'implanter diverses normes de système d'exploitation. Les programmeurs nippons, habitués à la norme ITRON[40] peuvent utiliser eCos selon cette norme. D'autres programmeurs préférent coder directement en C. Enfin, la norme POSIX pourrait être aussi implantée dans le cadre d'eCos.

Le développement d'eCos est soutenu par douze géants de l'électronique : ST Microélectronique (Europe), ARM (Europe), Motorola (USA), NEC (Japon), OKI (japon), Panasonic (Japon), Toshiba (Japon), Toyota

40 http://tron.um.u-tokyo.ac.jp/TRON/ITRON/home-e.html

(Japon) ainsi que Cogent, EBS, Palmchip et Zentek.

Ce que l'on ne peut pas encore faire

Les logiciels libres pêchent par la faiblesse de l'offre en progiciels de gestion. Même chose pour les applications métiers, mettant en jeu des acteurs très atomisés : mise en page pour la presse, montage vidéo (voir Chapitre 6).

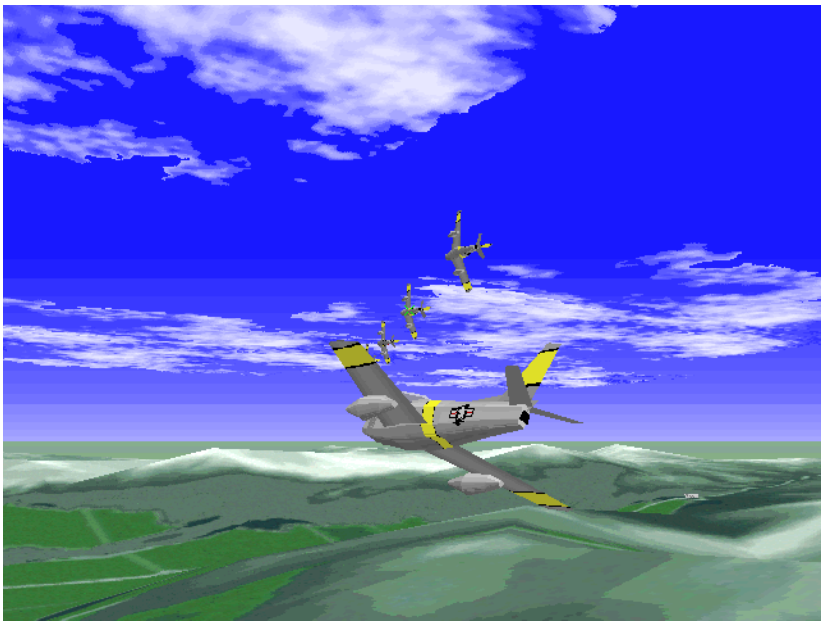

Figure 6. Simulateur de vol sous Linux (Sabre)

Côté grand public, la plate-forme Linux souffre de l'absence de versions de CD-ROM de jeux ou éducatifs. Certains jeux ont été portés (Doom, Quake, Abuse, SimCity etc.) voire développés sous Linux mais ils sont encore rares. Ces lacunes peuvent toutefois être surmontées par l'utilisation d'une émulation Windows ou Macintosh, ou d'une configuration *dual boot*, en attendant qu'un grand acteur (organisme public, grand

compte) ou un bénévole indépendant développe une offre de référence dans ce domaine.

EN BREF – Fiabilité, interopérabilité et richesse applicative font des solutions autour de Linux et des logiciels libres une alternative crédible à l'offre propriétaire dominante (Microsoft et Apple). Leur gratuité, ou leur prix peu élevé, font de ces outils une solution avantageuse pour les PME, les établissements d'enseignement, les instituts de recherche ou les collectivités locales, notamment pour l'informatique en réseau, les serveurs d'Intranet ou les postes bureautiques individuels.

Mais les logiciels libres souffrent parfois de leur passé universitaire : certains outils nécessaires aux entreprises comme les progiciels de gestion ou les jeux éducatifs pour le grand public sont encore absents de cette offre. Toutefois, la possibilité laissée aux individus de développer ou d'enrichir l'offre logicielle comble peu à peu ces manques. Qui plus est, la présence croissante des logiciels libres dans l'entreprise ne laisse pas indifférents les grands acteurs du marché qui comblent rapidement les lacunes applicatives des logiciels libres par une offre propriétaire complémentaire.

4

Chronique d'une adoption

Linux et Apache ont aujourd'hui acquis des parts de marchés très importantes dans le domaine des serveurs (Intranet, réseau local, serveurs d'applications, etc.). Internet, marché lui aussi en plein boom, est dominé par les outils libres. Les éditeurs commerciaux ne peuvent donc plus négliger ces gisements de croissance. A partir de l'été 1998 et l'intégration d'Apache à l'offre Websphere d'IBM, la vague Linux a touché la plupart des grands acteurs de l'industrie. Au passage, les éditeurs en ont profité pour faire jouer la concurrence et s'affranchir de la tutelle parfois étouffante de Microsoft.

Apple fait son Unix

Aussi surprenant que cela puisse paraître (rien de plus éloigné d'un logiciel libre que MacOS), Apple a été l'un des premiers éditeurs commerciaux à développer officiellement une offre autour de Linux. La raison en est simple : Apple avait de fortes parts de marché dans l'éducation, no-

tamment dans les universités. Il avait donc besoin d'un Unix pour le monde de la recherche. Pour les processeurs 68000, la firme à la pomme avait déjà développé un Unix propriétaire, baptisé A/UX. Mais, lors de la transition au processeur PowerPC, un dilemme se posa : fallait-il développer un nouvel Unix ou en acheter un, par exemple, l'AIX d'IBM qui fonctionnait déjà sur cette plate-forme?

Apple a choisi une troisième voie : avec la participation de l'Open Sofware Foundation[41], la firme a participé au financement du développement de MkLinux, une version de l'OS libre qui utilise le micronoyau MACH, que beaucoup considèrent par ailleurs comme la base de son futur MacOS X. Des antiques modèles Nubus aux derniers G3, n'importe quel ordinateur Apple peut fonctionner avec cette distribution. Mais il n'en reste pas moins que Linux n'est pas vraiment au cœur de la stratégie d'Apple.

Netscape, le retour aux sources

L'ouverture du code source de Communicator par Netscape a eu un tout autre retentissement. Il ne s'agissait pourtant que d'un retour… aux sources. A l'origine, Netscape Navigator a été développé à partir de Mosaic, un logiciel semi-libre qui était livré avec son code source mais était payant si l'on en faisait un usage commercial. Or Navigator était commercial et son code source n'était pas disponible.

L'arrivée d'Internet Explorer, qui était gratuit et a fini par devenir presque aussi performant que Navigator, a donc poussé Netscape à un revirement stratégique en janvier 1998. Pour rester compétitif, il fallait que son navigateur reste meilleur et ait davantage de fonctions que celui de Microsoft. Or, ces développements exigeant des investissements im-

41 L'Open Software Foundation est un consortium de constructeurs qui développe notamment la bibliothèque graphique Motif destinée au monde Unix.

portants, l'éditeur a choisi de revenir au modèle Mosaic.

L'ouverture du code permet de laisser à Mozilla, la communauté des développeurs de Communicator (la suite cliente qui intègre Navigator), le soin d'effectuer les améliorations qui seront intégrées par la suite au produit. Grâce à sa licence semi-libre, la NPL (voir chapitre 5), Netscape peut réutiliser ces développements dans ses serveurs. A un moment où il était en difficulté face à Microsoft, Netscape a pu réduire ses charges de développement sur Communicator en se consacrant davantage à son offre serveur. Cette adoption, initiée par le co-fondateur de la société Marc Andreesen après la lecture de « *the Cathedral and the bazaar* », semble toutefois être davantage qu'une simple stratégie opportuniste. Fin 98, Netscape a en effet aussi livré gratuitement son *messaging server* et son *directory server* sur Linux.

Corel lance le Netwinder

Pour Corel, davantage encore, l'alliance nouée avec le monde du logiciel libre prend place dans le cadre d'une stratégie de survie. Wordperfect est certes disponible depuis longtemps sous Linux et peut être utilisé avec NetBSD, via la compatibilité SCO. Mais c'est après l'échec de sa stratégie de développement d'applications en Java pour les *Network Computer* (NC), que Corel a décidé de miser sur Linux. Le Canadien a donc développé, autour d'un processeur ARM, le Netwinder, une solution client-serveur de type NC entièrement sous Linux. Au lieu de restreindre son usage aux applications développées pour une machine virtuelle Java, le Netwinder est capable de faire tourner des applications Linux natives et permet d'accéder aux applications de l'entreprise hébergées sur un serveur central.

Corel s'est encore engagé davantage en portant la version 8 de WordPerfect sous Linux et en l'offrant gratuitement en téléchargement

aux particuliers et aux établissements d'enseignement. A l'image des éditeurs de distributions Linux (voir Chapitre 6), l'éditeur se rétribue en vendant le support technique et le manuel d'utilisation.

Adaptec

A l'été 1998, les utilisateurs de Linux ont aussi reçu un soutien décisif pour leur plate-forme, de la part d'un fabricant de matériel. Adaptec, l'un des principaux fournisseurs de cartes SCSI pour serveurs, a annoncé qu'il allait développer des pilotes pour périphériques sous Linux. Il était ainsi le premier à reconnaître l'existence d'un marché important pour l'Unix libre.

Auparavant, la plate-forme souffrait d'un handicap de taille : les fabricants de cartes refusant de livrer les spécifications de leurs produits, il était très difficile aux développeurs de porter les pilotes de périphériques sous Linux.

IBM danse avec les fantômes

Comment faire lorsqu'on s'appelle IBM, que l'on mise sur le commerce électronique, mais que l'on ne veut pas avoir à utiliser l'offre d'un grand concurrent comme Microsoft? A l'été 98, la firme d'Armonk a résolu le problème en réalisant l'incroyable : intégrer à son offre d'*e-business* WebSphere le serveur libre Apache.

Pour IBM, ce choix a d'abord eu des allures de casse-tête : comment incorporer à ses packages un logiciel qui n'est pas régi par le droit d'auteur traditionnel ? « Même si on voulait payer, il n'y avait personne à payer. Ils (les auteurs) n'existent pas, légalement », s'exclamait Yen-Pin

Shan, l'un des programmeurs de l'entreprise dans Forbes[42]. Autant négocier avec des fantômes. Pour tout interlocuteur, les commerciaux d'IBM n'avaient en effet qu'un groupe d'une vingtaine de programmeurs répartis aux quatre coins du globe, de Palo Alto à Munich. Les décisions concernant le serveur était prises par vote. Enfin, le logiciel étant sous licence libre, il n'était pas à vendre.

En échange de l'intégration à Websphere, IBM a donc accepté de participer au progrès technologique du produit, en le portant sous AS/400 ou en ajoutant son accélérateur de cache à réponse rapide (FRCA) et la sécurisation SSL à la version NT du serveur.

IBM renforce d'ailleurs pas à pas son engagement puisqu'il prévoit de porter son gestionnaire de bases de données DB2 et sa suite d'applications de productivité Lotus Smart Suite sur Linux, ainsi que ses outils de groupware Notes et Domino. Le code source de Smart Suite pour Linux devrait même être disponible gratuitement. Fin 98, IBM a mis sur le marché une version Linux de son système de fichiers en réseau AFS et publié sous forme de logiciel *open source* un serveur de courrier électronique sécurisé (Secure Mailer, appelé aussi Postfix). En France, IBM a offert aux établissements scolaires de Strasbourg des serveurs sous Linux.. Début 99, des informations[43] circulent à propos de son arrivée imminente sur le marché du service Linux avec des serveurs PowerPC préinstallés.

La ruée des SGBD

Il existe depuis longtemps des bases de données commerciales sous Linux, telles Solid, D3 de Pick Systems et Adabas de Software AG. Pourtant, l'annonce par Oracle[44] de la disponibilité de son système de gestion

42 « For the love of hacking ». Forbes, 10 août 1998.

43 http://www.techweb.com/se/directlink.cgi?CRN19990208S0005

44 Pour une critique d'Oracle 8 sous Linux, voir Le Monde Informatique du 23 octobre 1998, sous le titre « Oracle 8 roule vite mais simple » . Traduction de « Oracle 8 shows on Linux promise »,

de bases de données relationnelles (SGBDR) « Oracle 8 » sous Linux a fait l'effet d'une traînée de poudre chez les grands éditeurs de gestionnaires de bases de données.

Informix avec Dynamic Server, Sybase avec Adaptive Server et Computer Associates avec Ingres ont tous suivi en l'espace de quelques mois. Les versions Linux de ces serveurs de renommée mondiale sont désormais téléchargeables gratuitement.

Création et développement

Sur le Web, seules quelques rares catégories d'applications souffraient encore de lacunes dans l'offre Linux. Or Golive, l'éditeur de l'un des logiciels de création de pages Web les plus réputés, a annoncé la disponibilité d'une solution complète d'édition de grands sites. Elle intègre des fonctions de *workflow* et le SGBD PrimeBase pour Linux, ce qui en fait l'un des premiers logiciels commerciaux à mettre à profit la possibilité de gérer des bases de données sous Linux accessibles par le Web.

Côté serveurs Web d'applications, Allaire a annoncé que son produit Cold Fusion, l'un des leaders sur son marché, supporterait Linux. L'environnement Codewarrior de l'éditeur Metrowerks, l'un des outils de développements les plus utilisés sur plate-forme Macintosh (mais aussi sous Windows) sera disponible sous Linux au printemps 1999. Parallèlement, IBM a aussi annoncé le portage du générateur d'applications propriétaire VisualAge Pacbase (Batch, TP, client-serveur) puis publié un compilateur Java hautes performances (Jikes) sous forme de logiciel *open source*.

Infoworld, 5 octobre 1998.

Linux fait Java

Ayant tous deux vocations à interopérer de manière universelle, Linux et Java devaient bien finir par se rencontrer. Sous la pression de ses clients, notamment universitaires, Sun a donc fini par annoncer le portage sous Linux de son Java Development Kit (version 1.2). Même chose pour Corba que l'Object Management Group (OMG) a porté sous Linux.

Parallèlement, Sun a aussi choisi de rendre public le code source de Jini, un logiciel qui fait communiquer toutes sortes de périphériques, tout en conservant la propriété intellectuelle, grâce à un montage juridique comparable à la licence publique de Netscape. Mais Java n'était toujours pas un logiciel libre alors que la société d'origine anglaise Transvirtual proposait une machine virtuelle compatible Java sous GPL et que Cygnus lançait un projet de compilateur Java natif. Aussi, Sun a finalement accepté d'appliquer à Java le même modèle que celui de Jini. Java fait donc désormais partie des logiciels libres ou semi-libres.

Les constructeurs brisent le tabou

Du fait des accords commerciaux conçus par Microsoft pour obliger les grands constructeurs, en échange d'un rabais, à payer une licence pour chaque micro-ordinateur indépendamment du nombre de copies Windows installées ou réellement utilisées, ceux-ci se risquent rarement à installer des systèmes d'exploitations libres par crainte d'annulation du rabais. Qui plus est, cela les contraindrait à investir dans la mise en place de chaînes spécifiques, tout en continuant à payer, en sus, la licence Windows. Aussi, seuls des petits assembleurs de PC comme VA Research[45] proposaient l'installation par défaut de Linux ou de FreeBSD sur un ser-

45 http://www.varesearch.com

Au Japon, l'Empire du Soleil Linux

Le concept de licence utilisateur n'est guère adapté à la culture japonaise. On a du mal à séparer l'idée de logiciel de celle de matériel. Lorsque les premiers dictionnaires électroniques sont apparus dans les années 90, les Japonais les ont intégrés dans des appareils électroniques spécialisés et non sous forme de logiciels pour micro-ordinateurs..

La notion de logiciel libre a donc connu une forte adoption de la part des fabricants. eCos, l'OS libre conçu par la société Cygnus pour les systèmes embarqués, est soutenu par la plupart des industriels japonais de l'électronique. Plus récemment, Konica a sorti le Qscan, un scanner de diapositives fourni avec des pilotes Linux et FreeBSD, compatibles notamment avec le logiciel de retouche d'images Gimp.

Plusieurs sociétés se sont lancées sur ce créneau. Le VAR Otsuka Shokai commercialise ainsi les serveurs Netfinity d'IBM Japan et les Express 5800 de NEC avec Linux préinstallé. Parallèlement, les logiciels libres titillent aussi un autre penchant de la culture japonaise, le *kaisen* ou esprit de coopération pour une amélioration par petites touches. La distribution japonaise Turbo Linux de la société Pacific Hi-Tech, s'est ainsi davantage en Septembre 1998 que MacOS dans l'Archipel. L'élève a réussi à dépasser ses maîtres américains, puisque c'est cette distribution qui a été retenue par Oracle pour son offre Linux.

veur ou un poste de travail.

Mais cette barrière a été peu à peu levée par la montée de la vente directe et de la configuration à la demande. Siemens fournit depuis 1997 SuSE Linux en standard sur certaines stations de travail. Aux Etats-Unis, Dell propose depuis un an, des PC et serveurs sous Linux, dans le cadre de ses contrats de fabrication à la demande[46]. Le constructeur a d'ailleurs signé un contrat de de service à valeur ajoutée (VAR) avec Mandala qui est chargé d'effectuer l'installation de l'OS[47].

Comme les éditeurs de SGBD durant l'été 1998, la plupart des autres grands constructeurs ont annoncé au début de cette année une offre intégrant Linux. Fin janvier, Hewlett-Packard s'est engagé sur Red Hat Linux

46 « Dell ships PCs, servers with Linux ». Interactive Week, 8 septembre 1998.
47 « Mandala et Dell unis pour Linux ». Distributique, 29 octobre 1998.

pour sa gamme de serveurs Intel Net Server, tout comme Compaq qui va intégrer la distribution américaine à ses serveurs sous processeur Alpha et annonce le développement d'une version sous Linux de son logiciel de téléadministration d'équipement en réseau : WebJetAdmin. Et Siemens, livrera en mars des clients légers avec un noyau SuSE Linux.

Les grands investissent le business du « libre »

L'adoption des plates-formes par les industriels va de pair avec des investissements dans des *start up* dont le logiciel libre est le fonds de commerce. Intel et Netscape ont ainsi investi dans Red Hat, qui distribue la version de Linux homonyme, avec un œil sur le marché des petits et moyens fournisseurs d'accès Internet (serveurs Web d'entrée de gamme). Quant à la *start up* fondée autour de sendmail, le logiciel libre généralement utilisé pour transporter 70% du courrier électronique sur les artères de l'Internet, elle est soutenue par plusieurs gros capital-risqueurs américains.

Même chose pour France Télécom (via sa filiale d'investissements Innovacom) et NTT, le principal opérateur japonais de télécommunications, qui sont devenus actionnaires du constructeur Cobalt Micro qui distribue Red Hat Linux dans ses micro-serveurs. Enfin, les entreprises elles-mêmes éditrices de plates-formes Unix comme Digital-Compaq, SCO ou Sun ont rejoint le consortium Linux International.

…Et Microsoft réagit

Considéré comme le plus menacé par la montée en puissance de Linux, Microsoft a toutefois porté certains de ses produits vers l'OS libre comme, par exemple, son diffuseur multimédia Netshow. De son côté, Software AG a développé une version du modèle objet de Microsoft

DCOM sous Linux.

Mais en ces temps de procès antitrust, Linux a surtout été utile à la firme de Redmond pour se désigner un concurrent. Après que Bill Gates a assuré, lors d'un passage à Paris en septembre 1998, qu'autour de lui c'était le vide, avec seulement Sun et IBM pour faire de la figuration[48], son plus proche lieutenant Steve Ballmer affirme que le principal concurrent de Windows s'appelle Linux.

En novembre dernier, le « Mémorandum d'Halloween[49] » affirmait d'ailleurs que « la première menace contre Microsoft est celle de Linux contre NT. Les tactiques fondées sur la trilogie FUD[50] ne peuvent être utilisées pour le combattre ». Mais c'est surtout le modèle de développement des logiciels libres qui inquiète la firme de Redmond. L'ouverture systématique du code remet notamment en cause la collaboration étroite de Microsoft avec le milieu des développeurs, de plus en plus séduits par ce mode de fonctionnement.

En guise de contre-offensive, le rapport recommande d'une part d'étendre les protocoles utilisés par les logiciels de Microsoft afin de créer de l'incompatibilité et, d'autre part, d'utiliser le droit des brevets pour barrer l'accès aux marchés aux logiciels libres (voir chapitre 5). Il s'agit d'ailleurs là de l'une des menaces les plus sérieuses qui pèsent sur le modèle *Open Source*.

EN BREF – De tout ce qui précède, une conclusion se dégage, celle explicitée par Eric Raymond, l'un des théoriciens du logiciels libres, dans son texte fondateur, « la Cathédrale et le Bazar » : « Les logiciels libres ne sont pas l'opposé des logiciels commerciaux. Le contraire de libre (*open*), ce n'est pas commercial, le contraire de libre (*open*) c'est propriétaire (*closed*) »

48 Lors de l'IT-Forum organisé par IDC à Paris.
49 Dont l'authenticité a été reconnue par la firme de Redmond.
50 *Fear, Uncertainty, Doubt* (peur, incertitude, doute)

5

Le droit du Logiciel Libre

Le cadre juridique des logiciels est un élément déterminant pour la naissance et surtout, la survie du modèle *Open Source*. En fonction des pays, les logiciels sont protégés par divers droits parfois incompatibles ou incohérents. Les auteurs de logiciels sont protégés par le droit d'auteur, le droit du logiciel le droit des marques ou, dans certains pays, le droit des brevets. Les consommateurs-utilisateurs sont protégés en Europe par le droit européen de la consommation et le volet décompilation du droit du logiciel.

Le logiciel et le droit

Le droit d'auteur comme droit d'usage

Le droit d'auteur est notamment régi par la convention de Berne[51] au

51 Convention de Berne http://www.law.cornell.edu/treaties/berne/overview.html

niveau international et en France et par le code de la propriété intellectuelle. Il permet à tout auteur de logiciel de se déclarer comme l'auteur d'une œuvre originale et d'accorder un droit d'usage de son œuvre à une tierce-personne. Un logiciel acheté dans le commerce ne correspond donc pas à une vente d'un produit mais à la concession d'un droit d'usage. Notons que la plupart des contrats de licence aux Etats-Unis sont fondés sur le droit d'auteur pour définir la notion de royalties, rémunérer l'auteur, interdire les copies sans autorisation de l'auteur et lever toute forme de responsabilité de la part de l'éditeur.

La loi européenne favorise les formats ouverts…

En Europe, une directive du 14 mai 1991 a défini un droit du logiciel fondé sur la protection du contenu d'un logiciel par le droit d'auteur. Cette directive introduit par ailleurs un droit à la décompilation à des fins d'interopérabilité qui constitue un premier niveau de protection des utilisateurs contre les pratiques de fermeture des normes de certains éditeurs. Tout utilisateur peut exiger de l'éditeur d'effectuer les adaptations nécessaires à un logiciel pour le rendre compatible avec un autre. En cas de refus de l'éditeur, l'utilisateur peut décompiler le logiciel et l'adapter lui-même. La décompilation est interdite aux Etats-Unis à de rares exceptions près. Elle est autorisée au Japon, sauf pour les logiciels américains à la suite d'un accord bilatéral avec les Etats-Unis.

…Et empêche le dépôt abusif des brevets

Le droit des brevets est conçu pour protéger un procédé innovant susceptible d'applications industrielles. Les logiciels et les algorithmes ne bénéficient pas d'une protection en Europe par le droit des brevets alors que le Japon et les Etats-Unis les acceptent. Les brevets permettent en pratique de protéger les idées, les concepts ou les fonctionnalités d'un logiciel. Le Japon dépose environ 35.000 brevets chaque année sur ce type de produits et les Etats-Unis, environ 20.000. Par comparaison, la France

dépose chaque année environ 70.000 brevets tous domaines confondus. La valeur de ces brevets reste néanmoins incertaine tant il est difficile d'effectuer une recherche d'antériorité avant leur dépôt.

De nombreux brevets sur les logiciels sont donc annulés en cas de procès par la découverte d'une antériorité. Par ailleurs, la plupart des brevets sur le logiciel résultent plus d'une découverte d'une propriété mathématique, d'un algorithme ou d'une méthode d'organisation que de la mise au point d'un procédé industriel original et complexe. En assimilant les logiciels à des activités intellectuelles[52], la position européenne d'interdiction des brevets[53] est donc probablement la plus avisée.

L'obligation de résultat

L'application en Europe d'une directive du 25 juillet 1985 instaure une protection du consommateur contre les logiciels défectueux. Ce texte a été complété par la directive du 5 avril 1993 concernant les clauses abusives dans les contrats conclus avec les consommateurs. L'interprétation française donnée en août 1998 par le garde des Sceaux indique que « la responsabilité du fait des produits défectueux a vocation à englober la catégorie juridique des meubles à laquelle appartiennent les logiciels ». Les logiciels seraient donc bien, selon le droit de la consommation, des produits.

Il est donc possible de poursuivre un éditeur dont le logiciel serait défectueux et d'obtenir des dommages de la part de celui-ci. Ces directives ne s'appliquent qu'à la vente de produit à des particuliers ou à des professionnels non informaticiens. On considère par exemple que le directeur informatique d'une banque est un professionnel de l'informatique et que lorsqu'il achète la licence d'utilisation d'un logiciel sans garantie de fonctionnement, il sait ce qu'il fait et ne pourra donc pas poursuivre l'éditeur.

---

52 « La brevetabilité des inventions »; Frédéric Polland-Dulian, IRPI, ed. Litec, 1997.

53 Le système européen repose sur deux traités internationaux : la convention de Munich (1973) et la convention de Luxembourg (1975).

Au contraire, une PME qui achète un logiciel de comptabilité est considérée comme un consommateur et bénéficie donc de la protection accordée par les directives.

Dans tous les cas, les conditions du droit d'usage d'un logiciel doivent être clairement affichée sur l'extérieur de la boîte du logiciel et non, comme c'est souvent le cas, sur un carton à l'intérieur. Un logiciel présentant des risques de défauts et destiné à des professionnels doit donc être clairement présenté comme tel s'il est commercialisé dans un circuit de distribution accessible aux consommateurs.

Les marques

Le droit des marques permet à un auteur de déposer le nom d'un produit et d'interdire toute commercialisation d'un produit de même nom sur le même segment de marché. Ce dépôt est de portée nationale et doit donc être effectué dans chaque pays pour couvrir le monde. Il existe toutefois une procédure de dépôt communautaire.

Le dépôt de marque peut être facultatif pour les produits notoirement connus.

Incertitudes sur le droit des interfaces

Le principal domaine d'interprétation incertaine du droit du logiciel concerne les droits d'auteur sur les interfaces : interface avec l'utilisateur, interface entre deux logiciels etc. Des interprétations contradictoires ont été proposées aux Etats-Unis mais il semble qu'un consensus s'établisse pour exclure les interfaces du droit d'auteur. Ne serait-ce que parce que la diversité des interfaces est contraire à la standardisation et à l'intérêt général (imaginez que chaque constructeur automobile place le volant et les pédales à une position différente).

Tableau 4. Droit du logiciel en Europe, aux Etats-Unis et au Japon

	Europe	*Etats-Unis*	*Japon*
Décompilation d'un logiciel	Autorisé à des fins d'interopérabilité	Interdit	Autorisé
Brevet sur un algorithme, une norme ou un logiciel	Interdit par la directive européenne du logiciel	Autorisé[54]	Autorisé[55]
Droit d'auteur sur l'interface utilisateur	Interprétation incertaine	Interprétation incertaine	Interprétation incertaine
Responsabilité de l'éditeur	Oui	Non	Oui

Les grands types de licences

Le droit des logiciels libres est fondé sur le droit d'auteur. L'auteur, créateur du logiciel, concède à l'utilisateur un droit d'usage très large à travers un contrat de licence. Point commun à toutes ces licences : le droit de copier et d'adapter librement le logiciel. A partir de là, les points de vue et les licences divergent. Les licences de type « domaine public » autorisent une appropriation du code libre dans un logiciel propriétaire (ex. Apache). Les licences libres strictes interdisent cette appropriation (ex. GNU). Enfin, les licences nées avec la vague *open source* tentent d'interdire cette appropriation par quiconque hormis l'auteur initial du logiciel.

Les licences de type « domaine public »

Les auteurs de logiciels qui choisissent de placer leurs œuvres dans le domaine public abandonnent toute protection par le doit d'auteur ce qui permet à une société d'adapter et de modifier ces logiciels sous forme de logiciel propriétaire. S'inspirant du domaine public, les licences de type « domaine public » n'obligent pas non plus les utilisateurs à reverser les

54 Sous certaines conditions définies par des directives utilisées par l'Office des brevets.
55 Sous certaines conditions définies par des directives utilisées par l'Office des brevets.

modifications à la communauté.

La licence BSD a été conçue pour BSD (Berkeley Software Design), la famille des Unix libres développés à l'origine par l'Université Berkeley (Californie). Outre le fait d'autoriser la publication du code source et la copie gratuite, elle exige notamment la mention des auteurs du programme dans le code source du logiciel qu'il s'agisse d'une version d'origine ou modifiée. Par contre, la publication du code source des développements issus de BSD n'est pas obligatoire. Tandis qu'un *copyright* traditionnel peut être déposé sur ces produits, ce qui a pour conséquence le développement de variantes propriétaires à partir d'une version libre. Le système d'exploitation BSDi, dérivé de logiciels sous licence BSD, est par exemple commercialisé sous forme de logiciel propriétaire.

La licence X, conçue pour les logiciels du consortium X du MIT, reprend peu ou prou les principes de la GPL : « tout détenteur d'une copie d'un programme (nda : sous licence X) bénéficie des droits d'exploitation, de copie, de modification, de réutilisation, de publication, de distribution gratuite ou commerciale et reste libre de distribuer l'ensemble et/ou d'en vendre des copies en octroyant ces mêmes droits aux utilisateurs sous réserve d'inclure » un message stipulant l'absence de garantie du produit et dégageant le consortium X de toute responsabilité en cas de dommage[56]. Cette licence interdit en outre d'utiliser le nom X Consortium pour la promotion d'un produit dérivé sans autorisation écrite préalable. Le licence des produits Digital versés dans le domaine public suit des règles identiques.

Les licences de type « domaine public » ne garantissent donc pas l'absence de divergence propriétaire, c'est-à-dire de développement d'un logiciel propriétaire à partir d'une base de composants libres. Certains auteurs de logiciels libres considèrent ces divergences comme une forme de reconnaissance. La licence Apache incite même les sociétés éditrices de

56 Nous avons utilisé ici les traductions non officielles de René Cougnenc, reprises sur le site Linux France.

serveurs Web propriétaires à remplacer leur propre code par celui d'Apache. D'autres auteurs se sont au contraire sentis floués d'avoir contribué, par leur effort bénévole, à enrichir une entreprise commerciale. C'est ainsi que sont nées les licences libres strictes.

Les licences libres strictes

La Licence Publique Générale GNU (GNU General Public Licence) plus connue sous le nom de licence GPL a pour but de « garantir la liberté de partager et de modifier des logiciels libres et de s'assurer que ces logiciels sont effectivement accessibles à tout utilisateur[57] ». Créée en 1984 par Richard Stallman, fondateur de la Free Software Foundation (voir Chapitre 2), la licence GPL (*GNU General Public Licence*) régit aujourd'hui la grande majorité des programmes à code source ouvert dont Linux, les outils GNU, l'environnement KDE etc. Elle aura notamment évité leur appropriation par des tiers comme c'était le cas parfois avec les licences de type domaine public. Tout programme, à l'intérieur duquel il spécifié qu'il est régi par la GPL, peut donc être copié, distribué et modifié librement, à condition de préciser son origine et de le maintenir sous GPL. Un logiciel développé à partir d'un programme sous GPL devra donc être lui-même placé sous GPL, même s'il intègre des éléments nouveaux qui ne seraient pas régis par une licence libre.

Par exemple, un routeur ou un pare-feu intégrant Linux doit être livré avec le code source de l'OS mais aussi des développements effectués à partir de celui-ci ou, le cas échéant, avec un contrat garantissant de livrer le code source sous trois ans. Cette règle n'est pas toujours respectée, ce qui place certaines sociétés dans l'illégalité. « Il arrive parfois que nous ayons des problèmes avec des individus commercialisant des produits de la Free Software Foundation sans mentionner leur origine. Généralement, il suffit de leur envoyer nos avocats pour qu'ils interrompent leurs

57 Le but de la GPL, http://www.fsf.org

activités », a indiqué Richard Stallman lors d'une conférence donnée à Paris[58].

Par contre, la liberté de redistribution n'implique pas la gratuité. « Vous pouvez demander une rétribution financière pour la réalisation de la copie et demeurez libre de proposer une garantie assurée par vos soins, moyennant finances », indique le texte de la licence. Point essentiel : la conception et la distribution d'un logiciel sous GPL n'implique pas en effet l'obligation de fournir une garantie et l'auteur n'est pas tenu pour responsable des dommages liés aux dysfonctionnements de son produit.

La licence LGPL est destinée à certaines bibliothèques logicielles GNU. Il s'agit d'une version assouplie et non extensive de la GPL. En particulier, elle n'exige pas que les programmes utilisant des bibliothèques libres deviennent eux-mêmes libres. La FSF s'est en effet rendu compte que le caractère indistinct (du point de vue légal) des bibliothèques par rapport aux programmes qui les utilisent freinait l'adoption de ces sous-programmes lorsqu'ils sont libres. Les développeurs de programmes commerciaux tendaient en effet à refuser de faire tomber leurs produits sous licence GPL. Seuls les « travaux basés sur les bibliothèques » (*work based on the Library*) sous LGPL tombent sous le coup de cette licence, et non les programmes qui les utilisent (*work which uses the Library*).

La licence artistique offre à son auteur un plus grand contrôle sur les adaptations d'un logiciel tout en permettant à une tierce partie d'extraire une partie du logiciel pour l'intégrer à un autre logiciel, y compris à un logiciel propriétaire. Elle distingue la version libre standard d'un logiciel, sous contrôle exclusif de l'auteur, des versions libres modifiées que pourraient diffuser les utilisateurs. Les logiciels peuvent être diffusés sans code source. L'auteur d'un logiciel propriétaire intégrant, sous certaines conditions, une partie d'un logiciel libre sous licence artistique doit supprimer toute référence au nom de l'auteur et au nom original du logiciel.

58 Conférence donnée à l'université Paris 8 le 5 novembre 1998, à l'initiative de l'association April.

La licence Aladdin reprend la plupart des principes de la GPL en ajoutant des restrictions visant à empêcher de contourner la GPL et de vendre des produits propriétaires faisant appel de façon détournée à des logiciels libres. La licence Aladdin est utilisée pour la diffusion de l'interpréteur Ghostscript[59].

Les produits néo-libres et semi-libres

Le modèle GPL comporte un inconvénient majeur pour un éditeur comme Netscape. Si on intègre un logiciel libre à un produit commercial, la GPL exige qu'on livre aussi le code source du produit issu de cette fusion. Inversement, la licence BSD semble trop permissive.

Aussi, lorsque Netscape a décidé d'ouvrir le code source de Communicator, Nescape a conçu deux licences complémentaires : la licence NPL (Netscape Public Licence) et la licence MPL (Mozilla Public Licence). La NPL couvre le code de Communicator diffusé initialement par Netscape alors que la MPL et conçue pour les développeurs souhaitant proposer de nouveaux modules. La NPL et la MPL imposent la publication de toute modification du code source de Communicator sans empêcher pour autant d'intégrer à Communicator des modules propriétaires livrés sans leur code source.

Par ailleurs, Netscape est autorisé à utiliser certains développements dans ses serveurs sans pour autant être obligé d'ouvrir l'ensemble de leur code source. C'est ainsi qu'est né le modèle semi-libre, fruit hybride du *copyright* propriétaire et du *copyleft*. Ce type de licence a depuis servi de modèle à la plupart des entreprises qui, de Novell (NDS) à Sun, en passant par IBM, ont décidé de libérer le code de certains de leurs produits.

La licence SCSL (Sun Community Source License) de Sun, créée initialement pour Jini, le protocole Java de communication entre périphériques, reprend les mêmes principes que la NPL : ouverture du code

59 GhostScript permet de visualiser un fichier postscript et le le convertir dans divers formats dont PDF, TIFF. http://www.cs.wisc.edu/~ghost/index.html

source, droit d'usage, de copie et de modification. Mais, contrairement aux licences libres, Sun conserve un droit dans le domaine de la certification et l'étend aux distributeurs/intégrateurs. Du coup, Sun exige pour l'usage d'une de ses marque en cas de déploiement ou de commercialisation en entreprise le passage d'un test de certification à titre de garantie de compatibilité avec l'existant. L'éditeur exige par ailleurs des royalties auprès des tierces parties qui intégreraient un logiciel sous SCSL à leur offre commerciale. Par contre, pour un usage personnel ou éducatif, aucune certification ni paiement de licence n'est obligatoire.

Comme Sun, certains auteurs de logiciels souhaitent en effet restreindre la portée d'une licence libre à certaine catégories afin d'éviter notamment que des sociétés fassent des bénéfices en utilisant ou en distribuant des logiciels libres sans rémunérer l'auteur. Des produits comme MySQL utilisent de longue date ce modèle : ouverture du code source, modèle participatif et licences commerciales payantes pour l'utilisation de MySQL comme composant OEM. MySQL est néanmoins un logiciel libre régi par une combinaison de licences GPL, LGPL, BSD et de type Aladdin. Les tarifs proposés pour le produit ont, d'un point de vue légal, valeur d'indication et non d'obligation.

La société norvégienne Troll, qui édite la bibliothèque Qt pour le développement rapide d'applications portables sur tous les Unix et sous Windows souhaitait formaliser ce principe de paiement pour un usage commercial et d'usage libre dans le cadre d'un logiciel libre. Qt est en effet le seul produit qui assure le chiffre d'affaires de Troll. C'est ainsi qu'est née la licence QPL (Qt Public License). Elle autorise la copie, l'adaptation, ou la redistribution de la bibliothèque Qt mais limite son usage gratuit aux logiciels *open souce*, notamment aux logiciels diffusés sous licence GPL. Les auteurs d'applications commerciales et propriétaires faisant appel à Qt doivent s'acquitter d'un droit d'usage versé à Troll.

Conséquences

Pour les utilisateurs

La quasi-gratuité des logiciels libres en font d'excellents produits pour les grands déploiements ou l'intégration à des produits de grande diffusion

Les licences libres permettent aux utilisateurs de modifier le logiciel et de le diffuser, ce qui implique une réactivité accrue et une grande indépendance par rapport à la plate-forme. On peut corriger les bogues, proposer des améliorations ou des portages plus rapidement.

Ce modèle supprime le problème de la pérennité du fournisseur. Quand un utilisateur commence avec un logiciel libre, il pourra changer de prestataire ou se retourner vers un autre si celui-ci disparaît, le produit étant propriété de tous.

De même pour la pérennité des données : si l'on travaille avec un format propriétaire, il est possible de l'ouvrir dans la mesure où on dispose du code. On peut donc exporter des données d'un logiciel vers un autre, sans coût supplémentaire.

Pour l'industrie

Suivant les exemples de Netscape, Troll, IBM ou Sun, la diffusion par les éditeurs commerciaux de logiciels sous licence libre ou semi-libre devrait se généraliser car il s'agit du seul modèle de diffusion permettant à un éditeur d'éviter qu'un produit à succès ne soit cloné sous forme de logiciel libre par des utilisateurs souhaitant bénéficier de conditions d'utilisation favorables.

La diffusion d'un produit sous forme de logiciel libre permet par ailleurs d'en faire un standard et d'investir un marché de grande diffusion bien plus rapidement qu'avec l'approche propriétaire. Le but déclaré de la Jini Technology Public Licence créée pour le standard Java Jini « est

d'engager la communauté des utilisateurs et développeurs à aider à développer cette technologie et à favoriser la diffusion rapide d'applications » basées sur Jini[60]. L'objectif est encore plus transparent dans le cas du langage Java dont le code, régi par la SCSL, est désormais ouvert tout en restant protégé.

L'accusation de Microsoft qui voit dans cette stratégie une technique nouvelle de prédation, n'est pas sans fondement[61]. Un logiciel au code source ouvert, a fortiori s'il est appuyé par une grande entreprise, a de bonnes chances de devenir un standard contrôlé par une entreprise. Si la méthode est exactement inverse de celle de l'offre propriétaire, elle aboutit à un résultat identique. L'utilisation de Java à des fins lucratives reste en effet soumise au paiement de royalties et à l'autorisation de Sun, sous la forme d'une certification « 100% Java ».

Menaces légales sur le logiciel libre

L'usage du droit permet-il de limiter la progression des logiciels libres ? Certainement aux Etats-Unis où l'on brevette les logiciels et où l'on interdit généralement la décompilation. Probablement aussi au Japon où l'on brevette aussi les logiciels mais pas en Europe, malgré des dispositifs de protection du consommateur qui rendent illégales certaines dispositions des licences libres.

La GPL est-elle légale en Europe ?

Non, la GPL n'est pas tout à fait légale en Europe car elle exclut toute responsabilité de l'auteur sans pour autant préciser que le logiciel diffusé s'adresse à un public averti. Elle contrevient donc aux directives euro-

60 Jini est une technologie Java de communications entre périphériques, pour laquelle un premier projet de licence, la Jini Technology Public Licence, avait d'abord été rédigé, avant d'être finalement soumis à la Sun Community Source License.

61 Op. cit. (Mémorandum d'Halloween).

péennes sur la consommation ou sur les clauses abusives.

Pour rendre la GPL légale, il faudrait probablement remplacer la clause de non responsabilité de l'auteur par une clause indiquant un montant maximal raisonnable[62] de responsabilité comme, typiquement, le prix du CD-ROM si le programme est distribué dans le commerce. Par ailleurs, il serait judicieux que les sites de téléchargement indiquent systématiquement que les logiciels en téléchargement sont destinés à un public de développeurs ou de professionnels et ne peuvent être utilisés par un consommateur sans expérience.

Par extension, les éditeurs de distribution Linux devraient soit indiquer très clairement que les produits qu'ils diffusent ne sont pas destinés aux consommateurs, soit accepter de prendre la responsabilité de tout défaut ce qui nous semblerait tout à fait normal dans la mesure où les principaux éditeurs de distributions Linux jouent de facto un rôle de certificateurs de solutions.

Un logiciel libre peut-il devenir propriétaire ?

Oui, c'est possible à condition que tous ses auteurs prennent cette décision. Les logiciels libres relèvent en effet des œuvres complexes, c'est-à-dire des œuvres issues de la transformation d'autres œuvres. Aux termes de la loi du 3 juillet 1985, les droits moraux sont donc incessibles. Mais, dans la pratique, si un auteur fait cadeau de son produit à la collectivité, il serait considéré comme infondé qu'il veuille reprendre un logiciel dont il fait don.

Le droit d'usage conféré par le contrat de licence s'applique à tous les utilisateurs du logiciel mais pas aux auteurs eux-mêmes. Néanmoins, les utilisateurs ayant reçu le droit de copier et d'adapter un logiciel initialement libre peuvent continuer à diffuser le logiciel sous forme de logiciel

62 Attention : le droit de la consommation autorise un juge à revoir une clause limitative de responsabilité s'il s'agit d'un consommateur. Un juge peut requalifier la clause de responsabilité et relever le plafond.

libre. On assisterait alors à un phénomène de divergence : les auteurs initiaux diffusent leur logiciel sous forme de logiciel propriétaire alors que les utilisateurs initiaux poursuivent le développement d'une version libre.

Des protocoles secrets

« Le modèle du logiciel libre (OSS) n'a pris pied dans de nombreuses applications serveurs que du fait de la large diffusion de protocoles ouverts et facilement accessibles. En étendant les fonctions de nos protocoles et en en développant de nouveaux, nous pouvons interdire aux projets (de développement) de logiciels libres l'accès au marché », conseille le « Memorandum d'Halloween » de Microsoft.

Fort heureusement, le droit européen ou nippon autorise la décompilation des logiciels propriétaires. Il est donc tout à fait possible de retrouver les formats ou des protocoles rendus délibérément incompatibles et secrets par certains éditeurs de logiciels propriétaires. Notons que le droit nippon sur la décompilation ne s'applique pas sur le marché américain en raison d'un accord bilatéral nippo-américain faisant suite à de fortes pressions politiques. En revanche, la décompilation au Japon de logiciels achetés en Europe, y compris américains, est probablement légale.

Le brevet m'a « tuer »

L'arme ultime contre les logiciels libres, ce sont les lois sur les brevets qui permettent aux Etats-Unis et au Japon, pendant vingt ans, aux inventeurs d'un procédé d'en conserver le contrôle. Aux Etats-Unis, le brevet est d'ailleurs cumulable avec le droit d'auteur et le droits des marques. Les brevets sont appliqués depuis peu à certains formats de données ou à des protocoles de communication ce qui interdit le développement de logiciels libres compatibles.

Il s'agit d'un vrai danger pour les logiciels libres[63] mais aussi d'une op-

63 Lire sur le sujet des brevets les textes de la « League for Programming Freedom »

portunité pour l'Europe (voir Chapitre 9). Les seules actions envisageables pour résister à la pression des principaux promoteurs du brevet sur les logiciels (IBM, Apple, Sun, Microsoft etc.) consistent à :

- publier toute idée de procédé chez un tiers de séquestre[64] afin de constituer une base d'antériorité librement utilisable et éviter que des éditeurs ne s'approprient des idées triviales sous forme de brevets ;
- constituer un fond de brevets à partir des meilleures idées, limiter l'usage libre de ces brevets aux logiciels libres et attaquer tout éditeur attaquant un auteur de logiciel libre pour violation de brevet ;
- lancer une action de *lobbying* auprès de l'Union Européenne pour contrer celle des éditeurs américains qui cherchent probablement à imposer la brevetabilité du logiciel en Europe au détriment des éditeurs européens et des auteurs de logiciels libres.

Le brevet n'est cependant pas une arme totalement efficace. L'inventeur français des systèmes de navigation automobile n'a par exemple jamais pu obtenir gain de cause auprès des industriels nippons qui produisent aujourd'hui ces systèmes alors qu'il avait déposé un brevet. Les brevets sur les logiciels sont fréquemment annulés pour cause d'antériorité. Enfin, il est rare qu'une entreprise dépense des frais d'avocats importants pour attaquer un auteur de logiciel libre non solvable.

Seuls quelques grands éditeurs de logiciels propriétaires pourraient décider d'attaquer les auteurs de logiciels libres pour contrefaçon. Leur image en serait probablement très dégradée et l'issue du procès incertaine car les logiciels sont aussi régis, aux Etats-Unis, par le principe de la libre expression.

---

http://lpf.ai.mit.edu

64 Les procédures de séquestre les plus courantes sont : les notaires, l'enveloppe Soleau (55F à l'INPI), une banque, un envoi de lettre scellée, des sociétés de séquestre sur Internet, la sociétés des gens de lettres (240F).

Le droit d'auteur sur les interfaces

Les années 80 ont connu divers procès pour contrefaçon d'interface utilisateur[65]. Leur issue est en général incertaine car la protection des interfaces, comme la position des boutons de formatage dans un traitement de texte, est considérée comme contraire à l'intérêt général ou ne relevant pas de la protection d'une œuvre originale. Néanmoins, une action en justice contre un auteur de logiciel libre, clone d'un logiciel propriétaire, pourrait être déclarée recevable et conduire à la saisie du logiciel. Aux Etats-Unis, le *digital millenium act* permettrait même d'interdire immédiatement la diffusion sur Internet d'un logiciel libre soupçonné d'avoir violé un *copyright*. L'effort législatif européen pourrait aussi menacer les logiciels libres en étendant la propriété intellectuelle[66] aux données publiques et aux idées.

L'une des seules forces des auteurs de logiciels libres face à ce type d'attaque est le caractère diffus de leur contribution et de leur responsabilité au sein d'une large communauté internationale. Une attaque d'envergure mondiale est coûteuse et il est difficile pour un éditeur d'obtenir dans chaque pays un jugement favorable sur un sujet aussi aléatoire que les interfaces.

Concurrence déloyale

Si l'éditeur d'un logiciel propriétaire peut prouver qu'une partie des éléments de son logiciel a été utilisée pour concevoir un logiciel libre équivalent, il peut envisager d'attaquer les auteurs du logiciel libre. L'auteur du logiciel libre serait alors rendu responsable de l'ensemble du manque à gagner induit par la publication du logiciel libre. Ce type d'approche butte néanmoins sur quelques obstacles qui la rendent largement inopérante :

---

65 Lire aussi sur le sujet des interfaces les textes de la « League for Programming Freedom » http://lpf.ai.mit.edu

66 http://www.monde-diplomatique.fr/1997/02/QUEAU/7750.html

- le plaignant doit pouvoir faire la preuve de l'usage d'éléments de son logiciel ;
- le plaignant doit pouvoir identifier les auteurs, ce qui est très compliqué dans le cadre d'un logiciel sous GPL conçu de façon communautaire ;
- les auteurs identifiés sont probablement insolvables.

L'accord de Wassenaar

En novembre 1998, l'administration américaine affirme avoir convaincu ses partenaires d'interdire ou de contrôler les logiciels de chiffrement dans le cadre de l'accord de Wassenaar qui réunit 33 pays développés. Le volet chiffrement de cet accord vise principalement à garder intactes les capacités américaines d'espionnage économique[67]. Mais cet accord ne s'applique qu'aux logiciels commerciaux. Les logiciels libres issus de la recherche sont pour l'instant épargnés.

Les pressions américaines[68] continuent pour obtenir de pays européens comme la Finlande ou le Danemark, pourtant très favorables au respect de la vie privée, une extension de l'accord aux logiciels libres. Or, nous avons vu au chapitre 3 que des logiciels libres tels que OpenBSD ou ssh étaient fréquemment appliqués à la sécurisation des réseaux. L'accord de Wassenaar constitue bien une menace additionnelle pour la diffusion de logiciels libres faisant appel au chiffrement.

EN BREF – Les logiciels libres reposent sur le droit d'auteur en autorisant la copie sous certaines conditions définies par un contrat de licence. La brevetabilité des logiciels (USA, Japon) et le droit de la consommation sont les principaux obstacles juridiques au développement des logiciels libres.

67 Vincent Jauvert. Comment l'Amérique nous espionne. Le Nouvel Observateur N°1779. 10/12/1998.
68 Voir les sites http://www.fsf.org/philosophy/wassenaar.html ou http://jya.com/crypto.htm pour suivre les avancements de ce dossier

6

L'économie du logiciel libre

Quelle logique peut pousser un individu ou une entreprise à développer un logiciel de qualité puis à le promouvoir sans rien réclamer en retour ? Peut-on faire confiance à un logiciel gratuit développé par une communauté d'utilisateurs bénévoles ? A ces deux questions, nous répondons par l'affirmative en démontrant l'existence de mécanismes économiques rationnels qui assurent l'efficacité et la pérennité des logiciels libres.

Ces mécanismes diffèrent en fonction du stade où ils se placent dans le cycle de vie d'un logiciel : création, diffusion ou maintenance. Mais ils ont néanmoins tous un point commun : répartir la charge de recherche et développement, au *pro rata* de l'usage, entre tous les grands acteurs de l'industrie, qu'ils soient utilisateurs, développeurs ou sociétés de services. En retour, les différents protagonistes bénéficient de standards ouverts et d'une liberté de développer ou d'adapter qui se révèle stratégique dès qu'un logiciel touche à la compétitivité d'une entreprise. L'économie du logiciel libre est donc créatrice de richesse, par un mécanisme de donnant-donnant ou, pour utiliser un terme anglais, de *win-win*.

La genèse des logiciels libres

Certains logiciels libres sont le fruit de travaux de recherche que des chercheurs souhaitent soumettre à une évaluation la plus large possible ou souhaitent offrir à la communauté pour accélérer leur valorisation et favoriser d'autres innovations dans le cadre du développement d'un patrimoine scientifique commun. D'autres logiciels libres sont développés par des développeurs indépendants dans le cadre notamment d'une compétition pour la gloire internationale (voir Chapitre 7).

Mais les logiciels libres sont aussi, souvent et depuis longtemps, développés par des industriels pour une raison purement micro-économique : dépenser moins.

Dépenser moins

Lorsqu'un organisme ne peut trouver sur le marché un logiciel correspondant à ses besoins, il est forcé de le développer. Dans les années 80, des universités américaines ont dû développer elles-mêmes des logiciels pour accéder à Internet à une époque où les entreprises étaient peu enclines à adopter et acquérir ce type de technologie. John Norstad, ingénieur système à la NorthWestern University, a par exemple écrit le logiciel de groupes de discussion NewsWatcher pour Macintosh. En optant pour une diffusion libre du code source auprès d'une large communauté d'utilisateurs-testeurs, ce logiciel a pu être débogué et amélioré plus efficacement que s'il avait été gardé secret, confié à un service de valorisation ou diffusé commercialement.

Depuis dix ans, la société Cygnus édite le compilateur libre gcc du projet GNU, rebaptisé récemment egcs. Ce compilateur est considéré comme l'un des meilleurs du marché et a servi de cadre à une expérience prometteuse de développement en commun de standards pour les grands de l'industrie. Dans les années 80, les grands fabricants de microprocesseurs avaient besoin de fournir à leurs clients un compilateur capa-

ble d'exploiter les spécificités de leurs nouveaux produits. Plutôt que de développer chacun un nouveau compilateur pour chaque nouveau processeur, ils ont confié à Cygnus et à la FSF le soin d'adapter et d'optimiser gcc puis egcs aux nouveaux processeurs en laissant certains de leurs ingénieurs participer au développement de ce compilateur libre à leurs heures perdues.

Plus récemment, les industriels nippons, alliés à des partenaires américains et européens, ont financé le développement par Cygnus du système d'exploitation embarqué eCos sous forme de logiciel libre. L'informatique embarquée est en effet un secteur clef au Japon, à la fois pour les producteurs de microprocesseurs et pour leurs utilisateurs dans l'automobile, l'électronique grand public, l'électroménager, les machines-outils etc.

Les industriels nippons s'étaient dotée d'une norme, TRON, mais avaient choisi pendant dix ans de développer en interne une multitude d'implantations de cette norme ou d'utiliser des implantations propriétaires commerciales. Les grandes sociétés avaient tendances à développer en interne pour préserver leur indépendance sur un composant logiciel jugé stratégique. Les plus petites sociétés qui n'avaient pas le moyens de procéder ainsi optaient pour l'acquisition de licences sur un produit commercial. Les industriels nippons se sont alors rendus compte que les coûts de maintenance étaient plus élevés avec une dispersion de l'effort d'ingénierie que s'ils disposaient d'une implantation commune suffisamment paramétrable pour être utilisée par tous et suffisamment libre pour que chacun puisse y contribuer. C'est ainsi qu'est né eCos, le premier logiciel libre américain compatible avec la norme TRON[69].

Il ne faut pas non plus négliger l'impact potentiel d'un tel logiciel sur une politique industrielle. D'une part eCos permet aux industriels nippons de diffuser leur norme dans le monde et, éventuellement, de freiner l'adoption de Windows CE qui pourrait menacer certains producteurs de

69 « Le logiciel libre investit l'informatique embarquée » par Mirel Scherer dans industries et techniques No 800, janvier 1999.

micro-processeurs comme ce fut le cas précédemment avec Windows NT et le PowerPC. D'autre part, l'adoption d'eCos comme norme permet aux industriels nippons d'éviter de devoir former à nouveau leurs ingénieurs à un nouveau système d'exploitation et de préserver leur savoir-faire dans le domaine de l'informatique embarquée.

Diffuser une norme

Les politiques industrielles ou l'intelligence stratégique ont souvent été un autre moteur, cette fois-ci macro-économique, du développement de logiciels libres utilisés comme vecteurs de diffusion agressive d'une norme.

Le contribuable américain finance donc fréquemment, sous forme de programmes publics de recherche ou de contrats de défense, le développement de logiciels visant à imposer mondialement une norme considérée comme favorable économiquement ou stratégiquement aux Etats-Unis. L'exemple le plus connu est celui des protocoles de l'Internet dont le succès, ainsi que celui de l'industrie qu'il a engendré principalement aux Etats-Unis, est probablement lié à la diffusion massive de tous les outils nécessaires à sa mise en œuvre sous forme de logiciel libre. Par ailleurs, des sociétés américaines sous contrats publics diffusent dans le domaine public, avec leur code source, des composants logiciels censés garantir la sécurité d'un réseau. Ces composants, devenus très populaires et intégrés à des produits commerciaux réputés, se sont avérés, au fur et à mesure des analyses de leur code source, contenir plus de 200 défauts de sécurité liés à des problèmes de dépassement mémoire. Restent tous les défauts qui ont peut-être été délibérément introduits par la NSA[70] mais pas encore découverts dans un produit devenu la norme du marché.

La pratique récente de la normalisation des protocoles de l'Internet

70 La NSA est l'organisme chargé aux Etats-Unis de l'espionnage électronique à vocation économique et stratégique. La NSA est à l'origine du réseau Echelon et des tentatives de limiter l'usage des technologies de chiffrement fort par les pays européens dans le cadre de l'accord de Wassenaar.

est heureusement devenue plus transparente depuis la création du World Wide Web Consortium (W3C) qui réunit autour du MIT (Etats-Unis), de l'Université de Keio (Japon) et de l'Institut National de Recherche en Informatique et en Automatique (France) les principaux industriels des technologies de l'information et de la communication. La publication de nouvelles normes de l'Internet, définies de façon collégiale, est systématiquement accompagnée de la diffusion de composants logiciels libres développés par l'un des partenaires.

Plus généralement, une entreprise, un consortium ou une société souhaitant développer une norme ayant vocation à être très largement diffusée, a tout intérêt à fournir sous forme de logiciels libres les composants nécessaires à son adoption et à son intégration dans d'autres produits. Dernier exemple en date : la société GoodNoise a acquis les droits lui permettant de diffuser sous forme libre un logiciel de lecture de fichiers sonores MP3 afin de se prémunir contre une appropriation de cette norme par les grandes maisons de disque et de garantir la pérennité d'une offre fondée sur la diffusion numérique directe d'œuvres musicales. Le logiciel libre joue dans ce cas le rôle de joker contre une industrie du disque qui se sent menacée par la diffusion directe et la disparition progressive des intermédiaires.

Se faire connaître

La gloire d'un auteur de logiciel libre se transforme parfois en bénéfices financiers lorsqu'elle devient un argument publicitaire. Un étudiant ayant créé un logiciel remarquable pourra plus facilement créer son entreprise. C'est ainsi qu'est né Netscape. C'est aussi ainsi que Netscape a répondu à la concurrence de Microsoft en offrant son code source à des millions d'Internautes et à des milliers de développeurs. L'éditeur espère faire de son site l'un des passages obligés de l'Internet et continuer à financer l'effort de développement de ses logiciels par la publicité engendrée par les visites de son site. La société suédoise TCX, éditeur d'un ser-

veur de base de données très performant, offre le code source de son logiciel et vit par ailleurs des services de conseil qu'elle vend autour d'applications des bases de données.

Garantir l'égalité

Il ne faut pas oublier enfin que dans beaucoup d'Etats les informations, données et autres biens publics[71] doivent être librement accessibles sans péage et sans droits. En France, tous les textes légaux et les formulaires administratifs sont, par exemple, libres de droits. Il semblerait naturel que, par extension, les logiciels permettant de lire la loi, de remplir les formulaires d'impôts ou les demandes de subventions soient eux aussi libres de droits et dépourvus de péage. Ce n'est cependant pas encore le cas aujourd'hui. Un chercheur qui demande des subventions à l'Union Européenne est contraint en pratique d'utiliser la suite bureautique Microsoft Office, à l'exclusion de toute autre, pour communiquer avec les services administratifs traitant son dossier. Le logiciel libre et les formats publics d'échange ne sont probablement pas encore assez développés dans le domaine de la bureautique et de relations administration-usagers.

Un circuit de distribution structuré

Bien qu'il soit souvent disponible gratuitement, le logiciel libre est générateur de richesse et a même donné lieu à la naissance d'un tissu économique dédié.

71 Un bien public est un bien qui peut être consommé par un nombre quelconque de personne sans perdre ses attributs ou dont la consommation ne peut être soustraite à quiconque à partir du moment où il existe. Voir sur ce sujet « Les biens collectifs », Alain Wolfelsperger, PUF 1969.

Entretien avec Bob Young, président et fondateur de Red Hat Inc.

Partisan dogmatique des principes du logiciel libre mais commercial retors, le Canadien Bob Young est un personnage représentatif de la jeune industrie de l'*open source*. Il nous a confié comment de grossiste en PC il est devenu vendeur de « marque » pour sa distribution Linux, Red Hat Linux.

« Pour moi, tout a commencé en 1994. A l'époque, je gérais ma propre société de distribution d'ordinateurs. En même temps, je rédigeais une *newsletter* pour le club des utilisateurs d'Unix de New York. J'ai demandé à mes lecteurs s'ils voyaient un sujet qui ne serait pas couvert par la presse informatique traditionnelle. Un grand nombre m'ont répondu : « On ne parle jamais de Linux. » C'est là que j'ai vraiment découvert le phénomène.

Mais au départ, je n'aurai pas parié sur cet OS. Il me semblait que Windows 3.1 serait forcément concurrencé par un système d'exploitation alternatif, comme NextStep, mais pas par Linux.

En fait, je me suis rendu compte que Linux était beaucoup plus sérieux que je ne l'imaginais, ce n'était pas seulement le fruit des loisirs de jeunes *hackers* de quatorze ans mais un vrai boulot professionnel. Les « hobbyistes », ceux qui écrivent des programmes pour leur plaisir, les week-end, ne sont qu'une des catégories de participant aux projets libres. Leur travail est d'ailleurs d'un très haut niveau car il s'agit souvent de leur métier dans la vie. Mais à côté des hobbyistes, on trouve aussi les ingénieurs ayant besoin d'effectuer un développement spécifique pour une grande organisation, comme par exemple, la Nasa (projet Beowulf, de PC en cluster pour supercalculateur) et les sociétés qui vivent du logiciel libre, comme Cygnus.

Avant que je ne créée Red Hat, c'était la distribution Slackware qui dominait le marché. Il me semblait qu'il existait une forte demande pour un package plus simple à installer et à utiliser, et mieux documenté. Je me suis donc associé avec Mark Ewink, auteur de la distribution Red Hat Linux, pour lancer cette société. Il apportait la compétence, moi le sens du marketing. Le but de Red Hat est en effet de créer une marque, un label, comme Heinz pour le ketchup ou Evian pour l'eau minérale. Une marque, c'est : un nom, une technologie et une image uniques.

Mais la concurrence de Red Hat avec d'autres marques n'est pas destructrice parce que nous ne fonctionnons pas selon un modèle propriétaire. Grâce à l'ouverture du code source, SuSE peut profiter des innovations que j'apporte à ma distribution. Nous évitons donc la balkanisation d'Unix et la création d'un marché à royalties de type Windows ».

Voir le portrait de Bob Young paru dans Le Monde Informatique du 19 février 1999.

Editeurs

Première catégorie de sociétés évoluant dans cet écosystème : les éditeurs de distributions de Linux, (Voir « distributions » Chapitre 2). Ils sont d'abord nés de la nécessité de diffuser l'OS libre sous forme de CD-ROM, d'aider les utilisateurs ne connaissant pas Unix en simplifiant l'installation et en fournissant une documentation et au besoin d'assurer un support pour rassurer les entreprises.

Leur rôle s'est depuis étendu à la fourniture de l'ensemble des composants logiciels nécessaires autour de Linux, pour arriver à remplir un certain nombre de fonctions d'assez haut niveau, comme par exemple, celle de serveur d'Intranet. Il existe donc des distributions générales qui accompagne Linux de toutes sortes de produits, le meilleur exemple étant SuSE On trouve aussi des solutions adaptables à des niches, comme le calcul parallèle (Extreme Linux, intégrant Beowulf et Red Hat) ou le multimédia. Parallèlement, les éditeurs fournissent aussi des produits purement commerciaux pour Linux. Ainsi Caldera propose une version de Netware, le serveur de Novell, pour l'OS libre[72].

Enfin, les « distributions » jouent de plus en plus le rôle des marques : elles vendent une part de rêve et servent à rassurer l'utilisateur grâce à la notoriété qu'elles ont acquise. Les « labels » les plus connus sont SuSE, Red Hat, Open Linux de Caldera, Slackware et Debian. En France, la société Les Logiciels du Soleil s'est aussi fait un nom en éditant une version francisée de la distribution Red Hat, commercialisée par Kheops.

Distributeurs

Deuxième catégorie : les distributeurs. Qu'ils soient constructeurs de matériel ou simples assembleurs, leur rôle est de concevoir une solution clef en main, comme un routeur ou un serveur d'Intranet. Il peut s'agir de solutions combinant matériel et logiciel tout comme de solutions pu-

72 Voir Infoworld du 17 août 1998.

rement logicielles. Exemple aux Etats-Unis : Cobalt Micro et son micro serveur Qube avec Red Hat Linux. En France, Mandala International[73] propose Linux Enterprise Server, une solution libre de serveur de réseau, comprenant notamment une émulation Mac/Windows/Netware, le tout pour 3 000 F HT.

Les constructeurs de PC jouent un rôle important en tant que distributeurs de solutions Linux. Ainsi, Dell vend des machines intégrant Linux dans le cadre de son programme de fabrication à la demande Dell Plus. Siemens Nixdorf fait de même en Allemagne en partenariat avec SuSE. Certains assembleurs de second rang centrent leur activité sur ce type de solutions, comme VA Research qui a signé un partenariat pour intégrer le SGBD d'Oracle dans ses serveurs Linux.

VAR

Deuxième catégorie : les VAR (*Value Added Reseller*). Dans le contexte des logiciels libres, leur rôle consiste à adapter une solution libre à une problématique dans une société donnée, puis à la déployer et en à assurer la maintenance.

En France, de petites SSII comme Alcôve (1,2 MF de CA en 1997) ou ESC Informatique (8 MF) se sont spécialisées de longue date sur ce secteur, s'engouffrant ainsi sur un créneau laissé vide par les grandes sociétés de services. Alors qu'ESC travaille surtout avec le monde de la recherche, Alcôve a décroché plusieurs grands comptes, comme l'Oréal, Les Trois Suisses ou Rhône-Poulenc. Cette percée dans les grandes entreprises lui a valu récemment de signer un accord de partenariat avec la SSII SQL Ingénierie.

Plusieurs *start up* prometteuses comme Kheops (créé par le fondateur des Logiciels du Soleil), Mandala International ou Atrid ont aussi fait leur apparition l'année dernière sur la scène des VAR Linux. Toutes vendent

73 Voir « Computer Reseller News » du 3 décembre 1998.

Une start up française: MandrakeSoft

- Comment avez-vous été amené à concevoir Mandrake ?

Gaël Duval, fondateur de MandrakeSoft: « C'est assez simple: j'aimais bien la simplicité et la puissance de la distribution RedHat (notamment le format de packages RPM), tout en étant frustré car il me semblait qu'il manquait à cette distribution une quantité de logiciels intéressants, et un véritable bureau graphique afin d'en simplifier l'administration et l'utilisation. Or ce bureau graphique existait, c'était KDE...

J'ai intégré tout ça, rajouté quelques fonctionnalités (par exemple, une meilleure gestion du clavier français, ou la possibilité pour un utilisateur non privilégié, de monter lui même le lecteur de disquette ainsi que le CD-ROM) et la deuxième version (5.1 - correspondant à la RedHat 5.1) a été rendue publique sur FTP le 23 juillet 1998.

A ma grande surprise, le succès a dépassé mes espérances: quand je suis rentré de vacances vers le 15 août, plus de 300 e-mails liés à Mandrake m'attendaient et trois distributeurs (deux américains, un australien) m'annonçaient qu'ils avaient commencé à vendre des CD-ROM Mandrake et que ça marchait bien !»

- Qu'avez-vous fait avant de fonder MandrakeSoft?

« J'ai fait mes études à l'université de Caen (licence, maîtrise, DESS d'informatique), puis un an et demi d'objection de conscience dans cette même Université. Après la sortie de ma méta-distribution, une société anglaise, puis une américaine ont voulu m'embaucher pour récupérer Mandrake, puis ayant la chance d'être assez impliqué dans le logiciel libre en France, j'ai eu pas mal de contacts qui m'ont permis de monter ma propre société (française !) pour continuer à faire vivre Mandrake. Je suis maintenant membre du directoire et employé de MandrakeSoft S.A. depuis le 15 décembre 1998. »

- Quel est le marché visé par MandrakeSoft?

« Nous espérons nous positionner comme une distribution Linux mainstream, la plus à même de séduire les utilisateurs actuels de Windows. Nous avons des accords d'édition pour notre PowerPack Mandrake dans plusieurs pays du monde, dont la France, les USA, la Russie et l'Australie. plusieurs autres sont en cours. Nous misons également dans un proche avenir sur le support technique Linux haut de gamme en entreprise. »

du conseil, du support technique, de la formation et de l'intervention sur site, par exemple, pour l'installation de Linux sur une machine. La plu-

part proposent aussi des offres packagées qui vont du serveur de fichiers à la station coupe-feu.

Certaines sociétés comme Mandala International, Alcôve ou Easter Eggs effectuent même des développements autour de logiciels libres. Mandala a porté certains pilotes de périphériques pour Dell France. Alcôve a réalisé un serveur de terminaux codes-barres en communication radio (Radio D) et un serveur de télécopie avec interface Web (Alfax Web), tous deux sous Linux. Quant à Easter Eggs, qui conçoit des routeurs et des serveurs Intranets sous Linux, il développe une solution simplifiée de commerce électronique, sous licence GPL, EEShop.

L'émergence de VAR spécialisés autour de Linux est liée au support croissant dont les grands éditeurs ont besoin dans le domaine des solutions libres. En effet, il est rare que ceux-ci disposent en interne des compétences nécessaires. Exemple : l'alliance technologique et marketing signé entre les éditeurs Red Hat, SuSE et Pacific HiTech et Oracle pour le support de son SGBD sous Linux.

La principale caractéristique de ce secteur est que, comme, tout appartient à tout le monde, la concurrence est très fluide. Etre une société de services dans le domaine du logiciel libre exige d'avoir des compétences - mais pas de moyens importants. Si bien qu'une petite société lorraine, Locabureau, a pu fabriquer un *firewall* avec serveur proxy en utilisant un PC auquel elle a intégré Linux. Après avoir vendu ce produit à quelques clients locaux dont l'administration régionale, elle envisage désormais de le distribuer plus largement comme une solution et passer du stade de VAR à celui de distributeur.

Des utilisateurs partenaires

La crainte la plus souvent exprimée à propos des logiciels libres par les utilisateurs de logiciels propriétaires est le manque de maintenance

dans la mesure où aucun acteur économique n'est clairement responsable du développement. Nous allons montrer ici, à l'aide d'un modèle simple, que la maintenance des logiciels libres est assurée et que son coût est réparti entre grands utilisateurs.

Le cycle économique des logiciels libres

On distingue deux types d'utilisateurs d'un logiciel libre : les utilisateurs actifs, qui participent à son développement, et les utilisateurs passifs qui se contentent d'utiliser des logiciels libres, de signaler les défauts et d'attendre les mises à jour. La viabilité des logiciels libres dépend principalement de l'existence d'une forte communauté d'utilisateurs actifs. L'existence de celle-ci est souvent garantie pour des raisons uniquement économiques.

Imaginons que vous ayez à déployer un logiciel dans votre entreprise. Evaluons les coûts de diverses solutions en fonction d'un modèle économique simple (voir figure). On considère trois scénarios :

1. acheter un logiciel propriétaire et payer des ingénieurs pour l'adapter à ses besoins ;
2. télécharger un logiciel libre et payer des ingénieurs pour l'adapter à ses besoins ;
3. payer des ingénieurs pour développer un nouveau logiciel (libre ou non).

Après que le logiciel a été déployé, des bogues vont probablement être découverts, tandis qu'il deviendra peut-être nécessaire d'ajouter des fonctionnalités. Ces défauts qui peuvent survenir plusieurs fois par an coûtent, chaque jour, à l'entreprise, en temps perdu par des utilisateurs qui doivent redémarrer leur machine, modifier des paramètres, saisir à nouveau des informations etc. L'entreprise doit donc agir vite pour réduire les coûts de ces défauts. Elle peut :

1. attendre une mise à jour du logiciel ;

2. payer des ingénieurs pour modifier le logiciel et corriger les défauts le plus rapidement possible.

Dans le cas d'un logiciel commercial propriétaire, il faudra régulièrement acheter une mise à jour si l'on choisit d'attendre ; il faudra acquérir une licence d'accès au code source si l'on choisit de corriger les défauts soi-même. Dans le cas d'un logiciel libre, les mises à jour sont effectuées par la communauté des utilisateurs actifs, ce qui signifie aussi que le coût de modification du logiciel est partagé par l'ensemble des utilisateurs actifs. Cette mutualisation des coûts est assurée par l'éditeur dans le cas d'un logiciel propriétaire, mais avec une perte de contrôle considérable, tant sur les prix que sur les priorités et la stratégie.

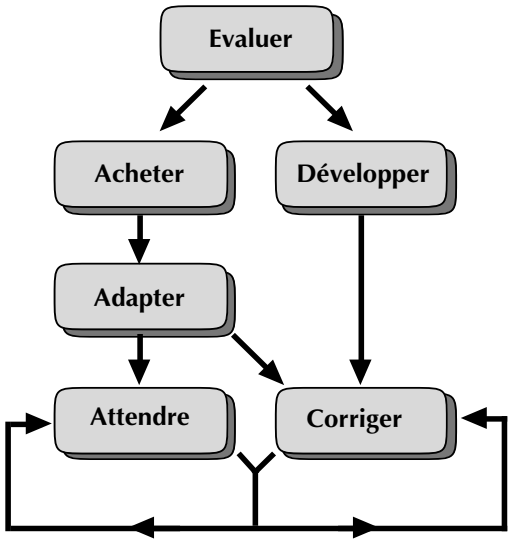

Figure 7. Le cycle économique des logiciels libres

Conséquences

L'étude détaillée de ce modèle[74] permet de déduire que :

74 L'économie des logiciels libres. Jean-Paul Smets-Solanes. Multimédium, Juin 1998. http://www.mmedium.com/dossiers/libre

- **Développer un logiciel libre est rentable lorsque l'offre commerciale est insuffisante**. En effet, dès que le coût d'acquisition des licences est important (par exemple s'il faut équiper plusieurs milliers de postes), lorsque le fournisseur met du temps à corriger les défauts, lorsque la technologie des logiciels existants entraîne des coûts importants d'adaptation ou lorsque les logiciels commerciaux ne sont pas fiables, il peut être rentable de développer un logiciel. En le rendant libre, on bénéficie alors de la réactivité des utilisateurs passifs qui signalent les bogues à l'avance et d'une communauté d'utilisateurs actifs qui partagent les coûts de maintenance. On retrouve donc ici l'argument économique qui a conduit au développement d'eCos.
- **Pour les projets stratégiques, le logiciel libre est un excellent choix.** En effet, l'accès au code source et la liberté de diffusion permettent d'éviter les risques inhérents à tout fournisseur de logiciel propriétaire : mauvais suivi si vous n'êtes pas un client prioritaire, refus ou prix élevé de l'accès au code source, changement soudain de tarification etc.
- **La qualité des logiciels libres s'adapte aux utilisateurs les plus exigeants**. En effet, lorsque, face à un défaut, le délai de réaction de la communauté des utilisateurs actifs d'un logiciel libre coûte trop en dysfonctionnements à certains utilisateurs passifs, ces derniers ont intérêt à devenir des utilisateurs actifs quitte à modifier eux-mêmes le logiciel et à publier la modification. Ainsi, le nombre d'utilisateurs actifs croît, ce qui fait automatiquement diminuer le délai moyen de réaction jusqu'à ce que l'on atteigne un état d'équilibre, chaque utilisateur corrigeant le défaut qui compte le plus pour lui. Il existe donc un équilibre économique qui garantit un nombre d'utilisateurs actifs suffisant pour assurer la maintenance et satisfaire les utilisateurs les plus exigeants.

Il n'est donc pas étonnant qu'une étude statistique américaine sur les logiciels[75] ait conclu que les logiciels libres étaient souvent les plus fiables.

Un point de vue d'utilisateur passif

Comparons logiciels libres et commerciaux dans le cas d'un utilisateur passif, c'est-à-dire un utilisateur qui se refuse à corriger lui-même les bogues ou à modifier le logiciel.

On constate immédiatement que le logiciel libre, en supprimant le coût des licences, autorise une marge appréciable en termes d'adaptation, de réactivité et de fiabilité. Si l'on omet les paramètres réactivité et fiabilité, ce que l'on gagne en licences non payées peut servir à payer des ingénieurs pour adapter le logiciel libre aux besoins de l'entreprise et à former les utilisateurs. En ce sens, le logiciel libre, adapté sur place, est créateur de plus d'emplois locaux que le logiciel propriétaire importé.

Exemple 1 : serveur Web

Supposons que l'on souhaite installer un serveur Web. Des choix possibles sont par exemple le serveur Web Microsoft sous Windows NT et le serveur Apache, dont une version optimisée est fournie avec SuSE Linux. Ces deux serveurs sont similaires du point de vue des fonctionnalités, mais Apache est réputé plus fiable et ses bogues sont corrigées plus rapidement car la majorité des fournisseurs d'hébergement sur Internet utilise Apache et ne peut tolérer d'attendre la mise à jour d'un produit qui est au cœur de leur stratégie.

À moins d'un argument de poids, comme la pression d'un chef de service ou d'un client, il vaut donc mieux choisir Apache puisqu'il coûte moins cher au départ et moins cher chaque année.

---

75 Fuzz Revisited: A Re-examination of the Reliability of UNIX Utilities and Services. Barton P. Miller. David Koski Cjin. Pheow Lee. Vivekananda Maganty. Ravi Murthy. Ajitkumar Natarajan. Jeff Steidl. Computer Sciences Department. University of Wisconsin. 1210 W. Dayton Street. Madison, WI 53706-1685. http://www.cs.wisc.edu:80/Dienst/UI/2.0/Describe/ncstrl.uwmadison/CS-TR-95-1268

Exemple 2 : une base de données

Comparons maintenant deux petits serveurs de base de données ; considérons par exemple le logiciel propriétaire FileMaker Pro (MacOS ou Windows) et le logiciel semi-libre MySQL (Linux). L'interface utilisateur de FileMaker Pro est probablement l'une des mieux conçues pour le commun des mortels. MySQL offre une interface utilisateur plus rustique à travers le Web mais est utilisé par la plupart des fournisseurs d'hébergement aux Etats-Unis et a été élu meilleur base de données de l'année par CNET[76]. Adapter FileMaker Pro demande donc moins d'effort que MySQL pour définir une application d'entreprise. En revanche, MySQL est plus fiable (il résiste aux tests SQL les plus durs) et son éditeur, TCX, doit corriger les bogues plus rapidement que FileMaker Inc. pour satisfaire des clients plus exigeants car plus exposés aux attaques sur Internet.

Le choix du logiciel libre est le meilleur si le nombre de postes à équiper est important ou si l'application est stratégique. Le surcoût d'adaptation du logiciel libre par rapport au logiciel propriétaire sera probablement compensé par l'absence de licence et, le cas échéant, rentabilisé sur le long terme grâce à une meilleure fiabilité et une meilleur réactivité. En revanche, si l'objectif est de monter très rapidement et en un seul exemplaire une base de données moyennement fiable, mieux vaut choisir le logiciel propriétaire car il reviendra moins cher à adapter.

Exemple 3 : logiciel de mise en page de documents

Comparons maintenant FrameMaker, un logiciel propriétaire de mise en page scientifique sous MacOS, et KLyX, un logiciel libre de publication scientifique pour Linux. FrameMaker permet souvent de réaliser des mises en pages complexes plus rapidement que KLyX grâce à une interface élégante qui gère les modèles de mise en page, les feuilles de styles, le placement des images, les hyperliens, la séparation quadrichromique, la conversion HTML, etc. L'interface très fluide du MacOS et les nom-

76 http://www.builder.com/Business/1998Awards/ss08.html?st.cn.nws.rl.bl

breux outils pour le graphisme disponibles sur cette plate-forme renforcent le confort d'utilisation et l'étendue des fonctionnalités. Mais KLyX offre des avantages par rapport à FrameMaker pour l'automatisation du processus de rédaction et la gestion des formules mathématiques complexes.

FrameMaker et KLyX étant des produits de stabilité équivalente, le choix de l'un par rapport à l'autre dépend donc de l'importance relative du prix des licences et des besoins en fonctionnalités avancées qui, s'ils ne sont pas satisfaits par une interface limpide, sont assimilables à des défauts. Aussi, dans une logique de court ou de moyen terme, le logiciel libre KLyX ne sera choisi que si le nombre de postes devant être équipés est élevé, que les documents à taper sont relativement simples et uniformes, ou que la main d'œuvre est bon marché.

Exemple 4 : développement d'un produit industriel

La société bordelaise Lectra a envisagé il y a quatre ans environ deux systèmes d'exploitation pour développer son système de CAO textile : Windows et Linux. D'un point de vue économique, c'est Linux qu'il fallait choisir en raison de sa fiabilité. D'un point de vue commercial, c'est Windows qu'il fallait choisir en raison de sa notoriété auprès des grands comptes. D'un point de vue stratégique, c'est Linux qu'il fallait choisir car l'accès au code source est absolument nécessaire pour pouvoir adapter, le cas échéant, le logiciel système aux contraintes très particulières d'un produit utilisé en milieu industriel où le coût des bogues est très élevé. Il serait par exemple inacceptable qu'une bogue du système d'exploitation entraîne un arrêt du travail.

C'est donc Linux qui a été choisi. Ce choix pourrait être remis en question si Windows était plus fiable et si son code source était aisément disponible.

Exemple 5 : production multimédia

Dans le domaine de la production vidéo et de CD-ROM multimédia, il n'existe pas encore de solution intégrée et stable à base de logiciels libres. Ce type de solution est coûteux à développer ou à adapter alors que de nombreux produits propriétaires existent déjà et ne coûtent que quelques centaines de dollars.

Les salaires des producteurs multimédias sont élevés – donc le coût des bogues aussi – et les produits commerciaux sont très fiables. Comme la production multimédia intéresse essentiellement des professionnels pressés et peu habitués à développer eux-mêmes des logiciels complexes, il est hélas peu probable que se développe rapidement une communauté d'utilisateurs actifs de logiciels libres pour la production multimédia en dehors de certaines niches.

Ce constat doit cependant être nuancé par la probabilité que des étudiants et des programmeurs bénévoles extraordinairement créatifs décident, comme ce fut le cas pour Gimp, de produire un logiciel multimédia de qualité professionnelle sans motivation économique directe[77]. Certains professionnels du multimédia sont alors susceptibles de revoir leurs choix d'investissement si un logiciel libre leur permet de réaliser de économies, quitte à financer occasionnellement un développeur pour effectuer des modifications.

Les limites économiques du modèle libre

Attention au coût de migration

Il ne suffit pas de savoir que le coût d'entrée et le coût annuel d'une solution à base de logiciels libres est le plus bas si l'on doit, par ailleurs, investir des sommes considérables pour sortir d'une solution propriétaire

77 A Thrifty Photoshop Built for the Web by Michael Stutz - Wired News 17.Mar.98.PS http://www.wired.com/news/print_version/email/other/technology/story/10975.html

et migrer vers celle des logiciels libres. Il convient donc d'évaluer si les gains de fiabilité et de réactivité de la nouvelle solution libre, ainsi que l'économie des renouvellement des licences, permettent de rentabiliser le coût de sortie.

Les grands éditeurs ont bien compris l'intérêt pour eux du coût de sortie. Une application de *workflow* réalisée dans l'environnement Lotus Notes est par exemple très coûteuse à porter dans un autre environnement en raison de caractère propriétaire du langage Lotus Script utilisé pour définir les flux d'information. Ce langage n'est en effet disponible que dans l'environnement Lotus. Il existe bien des solutions équivalentes sous Linux (Perl, Tcl etc.) mais elles ne sont pas compatibles avec Lotus Script et nécessiteraient de redéfinir les flux d'informations dans un autre langage.

De même, une solution de base de données réalisée avec Microsoft Access, Visual Basic ou la technologie ASP nécessite pour fonctionner la quasi-totalité de l'offre bureautique Microsoft ce qui rend impossible le portage vers une solution à base de logiciels libres à moins de consentir un effort important de développement. Des outils sont bien sûr disponibles sous Linux pour faciliter le portage sans l'automatiser complètement pour autant.

Pas de certification des VAR

Comme toute entreprise commerciale, les sociétés qui commercialisent des solutions libres doivent gagner de l'argent. Or comment faire des profits lorsque :

4. Le produit est gratuit, et ne permet donc pas de réaliser de chiffre d'affaires sur la vente de licences.
5. Le produit est fiable et ne nécessite que peu de maintenance.

Beaucoup de sociétés essaient donc de vendre une solution très spécifique (*custom*), ce qui leur permet de facturer des travaux de développe-

ment ou des services d'installation : la journée/ingénieur revient alors entre 3 000 et 5 000 F. Au lieu de facturer la maintenance, certains prestataires essayent aussi d'externaliser la gestion du serveur.

De manière générale, les logiciels libres souffrent encore de l'absence de norme pour l'installation. Contrairement aux produits propriétaires par les acteurs dominants (notamment Microsoft et IBM), les logiciels libres ne possèdent pas de certification de la part du fournisseur (en l'occurrence les communautés de développeurs). Celle-ci est donc assurée par les éditeurs de distributions (Red Hat, SuSE, etc.). Red Hat Inc. a ainsi mis en place un réseau de revendeurs autorisés[78] et commence à définir des normes pour l'installation. En France, la société a confié aux diverses entreprises fédérées autour de Joël Bernier (Logiciels du Soleil, Kheops, Erian Concept, etc.), la distribution de sa version de Linux sur le marché français.

De leur côté, les distributeurs-assembleurs (Corel, Cobalt, Freegate, CXR etc.) cherchent à promouvoir par une marque une solution complètement normalisée, tant du point du vue matériel que logiciel, sachant que l'installation et la configuration sont réduites à leur plus simple expression et ne nécessitent pas de fortes compétences externes.

Conséquence directe de l'émergence de Linux dans l'entreprise, le besoin de certification et de standardisation risque de modifier la donne dans l'économie du logiciel libre. Face aux distributeurs-assembleurs, les petites sociétés de services qui opèrent sur le marché du logiciel libre comme VAR pourraient être reléguées vers des contrats de maintenance ou d'adaptation de solutions normalisées. La montée des solutions préinstallées pourrait aussi aboutir à la domination de nouvelles solutions fermées : il n'est pas certain que l'on puisse remplacer simplement un micro-serveur Cobalt par un PC Dell sous SuSE. Un logiciel propriétaire sous Linux n'offre pas nécessairement plus de garantie de compatibilité

78 Red Hat Software Training and Certification http://www.redhat.com/training/ et http://www.redhat.com/enterprise/

avec la concurrence que son équivalent sous Windows. Or, l'interface-utilisateur de configuration des microserveurs Cobalt est propriétaire. Même dans le domaine du logiciel libre, des possibilités de divergences existent : les distributions Linux de Red Hat et de SuSE incorporant des fonctionnalités qui leur sont propres. On pourrait imaginer qu'elles refusent un jour de s'intégrer. L'installation des marques serait alors le début de guerres intestines.

Rien de prouve que ce scénario va se réaliser. En particulier, le rôle des standards dans le processus de développement des logiciels libres par ingénierie distribuée a toujours permis d'éviter les divergences graves d'un logiciel libre. Le principe d'accumulation de logiciels libres par l'éditeur de distributions puis de sélection des meilleurs logiciels par les utilisateurs conduit les marques à copier mutuellement leurs offres à quelques mois d'intervalles. Mais le fait que les « marques » ne participent pas ensemble à l'élaboration de standards - tout en maîtrisant le processus de certification - constitue un problème de fond. Il faudra donc un jour que le monde du logiciel libre se dote d'organismes comparables à des institutions comme l'Object Management Group pour les modèles objet.

On peut aussi imaginer un modèle où les sociétés de services, aujourd'hui réduites au rang d'installateurs des grandes marques, pourraient continuer à apporter de l'innovation au monde de Linux. En recourant à des outils entièrement libres comme la distribution Linux Debian et à des associations professionnelles comme l'Aful en France, elles pourraient développer et faire connaître des solutions locales, voire les certifier. Il leur faudrait alors financer en commun un label, soit un nom et une notoriété, condition sine qua non pour que leurs solution pénètre les entreprises. Cette certification associative permettrait de combiner l'efficacité de la « marque » et de la démocratie.

Pas de financement sans gros utilisateur

Les éditeurs de logiciels propriétaires ne vendent pas les logiciels à un prix proportionnel aux nombres de postes. En fait, il y a des ristournes, allant jusqu'à la gratuité, et des prix de gros qui sont destinés à dissuader les grands utilisateurs d'utiliser ou de développer des logiciel libres. Le fait pour un grand groupe de faire appel à un bureau d'évaluation pour comparer logiciels libres et logiciels propriétaires devrait d'ailleurs conduire l'éditeur d'un logiciel propriétaire à baisser encore plus ses prix tant la comparaison entre libre et commercial lui est peu favorable d'un point de vue technique ou économique.

Ainsi, la suite bureautique MS Office serait proposée à certaines multinationales françaises à moins de 50F par poste alors qu'une petite PME doit payer environ 4000F. Dans ces conditions, probablement assimilables à du *dumping*, peu de grands groupes sont prêts à envisager d'acquérir un produit concurrent européen bien moins onéreux comme Star Office ou à participer au développement de la suite bureautique libre KOffice. Les grands groupes ayant un rôle de prescripteur, de nombreuses PME financent par leurs achats au prix fort la ristourne accordée au grands groupes dans un système globalement plus coûteux pour tous que le recours à des produits plus compétitifs ou au développement en commun de logiciels libres.

Plus généralement, lorsqu'aucun grand groupe n'est capable de financer le développement d'un logiciel libre pour que le marché dispose de solutions informatiques à un prix raisonnable, il est rare que des PME s'associent pour financer ensemble le développement au sein d'un consortium. La profession de la presse et de l'impression est un exemple typique de communauté professionnelle qui accepte, sans s'organiser de façon concertée, les augmentations de prix répétées et les problèmes compatibilité du logiciel de mise en page Quark XPress, devenu il y a cinq ans le standard du marché en raison de son caractère alors innovant mais

servant désormais à financer les déboires commerciaux de son éditeur dans le multimédia. Pourtant, il serait probablement bien moins onéreux pour cette communauté professionnelle de financer le développement d'un concurrent d'XPress sous forme de logiciel libre, ne serait-ce qu'en raison des baisses de prix qu'une telle initiative entraînerait chez Quark.

Aussi, l'une des rares approches envisageables pour financer le développement de logiciels libres utilisés par une multitude de professionnels sans compétences informatiques consisterait à vendre des logiciels sous contrat de licence propriétaire à durée limitée et de les diffuser sous forme de logiciel libre lorsqu'une nouvelle version est mise sur le marché, lorsqu'un quorum d'acheteurs a été atteint ou au bout de quelques années.

Parfois moins de fonctionnalités chez les « libres »

Nous avons vu plus haut que dans le domaine de la production multimédia, il n'existe pas de solution intégrée et stable à base de logiciels libres. Ce constat mérite d'être explicité.

Il existe en effet de nombreux logiciels libres pour le multimédia : Gimp (retouche d'images), Mesa (affichage 3D), POV (rendu d'images de synthèse), Moonlight (modélisation 3D), Multitrack (enregistrement audionumérique multipiste), XAnim (lecture audio et vidéo) etc. Mais ces logiciels ne gèrent pas certaines fonctionnalités avancées nécessaires dans un cadre professionnel.

La calibration colorimétrique, par exemple, n'est pas gérée par les logiciels libres. Les formats vidéo les plus avancés (ex. Sorenson) et les séquences composites (mélange de son, texte, image, vidéo, 3D etc.) ne sont pas non plus supportés. Le copier-coller ne fonctionne pas pour les données multimédias et comme chaque application utilise ses propres bibliothèques de routines et ses propres formats, l'intégration de données disparates dans un même document est quasi impossible. Quant au déve-

loppeur, il ne dispose pas d'une bibliothèque de routines libres lui permettant de développer rapidement une application multimédia capable de gérer à la fois la calibration colorimétrique, les séquences composites, l'accès aux périphériques, la synchronisation, la 3D, etc.

Au contraire, l'offre de logiciels propriétaires dans le domaine du multimédia est très riche et la concurrence a joué son rôle de moteur de l'innovation et d'ajustement des prix. Le graphiste professionnel, qui est souvent *free lance* et n'a pas l'habitude de développer ses propres outils, peut donc difficilement justifier économiquement l'utilisation d'un environnement de logiciels libres relativement pauvre alors qu'existe une offre propriétaire diversifiée, accessible et conçue depuis dix ans pour maximiser sa créativité et sa productivité.

Le développeur, qui doit produire rapidement une application utilisant toute la richesse du multimédia, choisira probablement la bibliothèque Quicktime car c'est la seule à couvrir tous ses besoins à la fois sous MacOS et sous Windows pour un coût de licence quasiment nul. Le développeur qui souhaite bénéficier d'une gestion de la couleur parfaite utilisera probablement ColorSync sous MacOS faute de concurrence sérieuse sous Windows.

Or, si ni les utilisateurs ni les développeurs d'applications multimédias n'ont d'intérêt à s'orienter vers une approche « logiciel libre », il est peu probable que de telles applications voient le jour à moins qu'un autre acteur favorise l'émergence de technologies libres pour la production multimédia et pour la calibration colorimétrique. Cet acteur pourrait être, dans le cas du multimédia, un grand laboratoire public ou l'industrie de l'électronique grand public qui est aujourd'hui menacée par la domination sans cesse croissante d'Apple et de Microsoft sur les technologies de diffusion vidéo par Internet .

Les autres acteurs envisageables sont les éditeurs de distributions, à condition qu'ils atteignent une taille critique. Il est en effet dans leur inté-

rêt de pouvoir fournir des fonctionnalités équivalentes à celles des OS propriétaires. L'un des développeurs de Red Hat est par exemple payé à temps complet pour participer à l'évolution du noyau de Linux et s'assurer de sa stabilité. Tandis que SuSE AG salarie à plein temps 4 employés pour coordonner la programmation des pilotes pour les périphériques vidéo. Moins connue, la société Kachina Technologies, qui distribue des CD-ROM et des machines sous Debian, aide la communauté des développeurs de cette distribution non commerciale à effectuer un portage sur le processeur 64 bits Ultra Sparc. Pourquoi alors ne pas envisager le développement par les éditeurs de distributions d'une technologie multimédia intégrée à partir des nombreux composants existants.

Face aux modèles traditionnels

L'industrie du logiciel connaît aujourd'hui trois principaux modèles de développement :

- **La prédation de marché**. L'objectif premier de l'entrepreneur est de vendre, non pas un logiciel, mais son entreprise. Pour valoriser son affaire, il s'approprie des technologies par des dépôts de brevets et un marché par un marketing agressif consistant le plus souvent à convaincre les utilisateurs d'installer un logiciel gratuit en lançant de grandes campagnes de publicité. La valeur de l'entreprise dépend du nombre d'utilisateurs séduits et de la difficulté pour un autre acteur d'entrer sur le marché. Cette approche ne tient pas compte directement de l'intérêt de l'utilisateur.
- **Le partage des investissements**. L'objectif de l'entrepreneur est de vendre au marché un produit packagé répondant aux besoins des utilisateurs et à un prix correspondant à une mutualisation de l'effort de recherche et développement. Cette approche doit

prendre en compte initialement l'intérêt immédiat de l'utilisateur mais peut dériver vers une rente de situation si l'éditeur choisit de fermer ses formats et d'augmenter ses prix.

- **Acheter du service**. L'utilisateur a tout le contrôle et le prestataire effectue tous les développements à la demande, mais c'est très cher. Si le contrat est bien négocié, l'utilisateur possède un droit d'usage du logiciel développé équivalent pour lui à en posséder la propriété intellectuelle.

L'économie des logiciels libres correspond à une combinaison des deux derniers modèles. L'effort de développement est partagé comme dans le second modèle et la propriété intellectuelle appartient à l'utilisateur comme dans le troisième modèle.

La possibilité laissée aux utilisateurs, par un abandon partiel de propriété intellectuelle, de développer eux-mêmes des logiciels libres permet de réduire les risques de dérives vers une rente de situation de l'éditeur en position dominante sans pour autant empêcher les éditeurs de logiciels propriétaires de financer leurs investissements par la vente de licences dans le cadre du seconde modèle. En revanche, lorsque les utilisateurs ont la possibilité de développer eux-mêmes des logiciels libres , les stratégies de prédation (premier type) fondées sur les brevets ou la protection des interfaces et qui conduisent à des tarifications abusives ne fonctionnent plus.

Ceci ne concerne cependant pas les stratégies de croissance de type *first mover* fondées exclusivement sur un marketing agressif. Il n'est donc pas étonnant que des fonds de capital risque investissent dans des *start up* nées du logiciel libre comme Sendmail, Red Hat, Cobalt, etc. en comptant obtenir, par la valorisation d'une marque, des croissances aussi spectaculaires qu'avec les *start up* traditionnelles fondées sur la valorisation de brevets.

EN BREF – L'organisation de l'économie des logiciels libres reprend les structures traditionnelles de la chaîne de valeur informatique en y ajoutant comme spécificité la possibilité laissée aux revendeurs et sociétés de services (VAR) de devenir concepteurs de solutions (distributeurs) ce qui conduit à renforcer la concurrence sur le marché.

Du point de vue de l'utilisateur, le développement ou l'utilisation de logiciels libres est avant tout un choix économique rationnel dans lequel sont comparés la rapidité de réaction d'un éditeur traditionnel, la réactivité de la communauté des utilisateurs de logiciels libres, les coûts des mises à jours de logiciels propriétaires, les coûts de développements à effectuer en cas de besoin sur le logiciel libre, etc. Dans le cas de logiciels stratégiques ou de produits de grandes diffusion, cette comparaison est souvent favorable aux logiciels libres dont la pérennité est garantie par la structuration de plus en plus solide de l'économie du logiciel libre. Aussi, un nombre croissant d'éditeurs à sera conduit à adopter le modèle *open source* pour maintenir leur position sur le marché.

7

L'informaticien citoyen

Autant que les modèles économique et juridique - qui rendent possible la perpétuation du système *Open Source* - l'organisation sociale qui règle la conception apparemment gratuite des logiciels libres explique le succès rencontré par ces projets.

Les laboratoires du *freeware*

C'est Eric Raymond qui a le mieux théorisé le fonctionnement des microsociétés de développeurs. Son manifeste « The Cathedral and the Bazaar »[79] décrit les différences de méthode entre les systèmes figés sur lesquels sont basés le développement des systèmes propriétaires (« la cathédrale ») et le modèle des logiciels libres où chacun peut participer à l'avancement du projet (« le bazar »). Ce texte a influencé la décision de l'ouverture du code source chez de nombreux éditeurs commerciaux, à commencer par Netscape avec Communicator en janvier 1998.

79 « The Cathedral and the Bazaar », Eric Raymond, http://www.opensource.org

Par la suite, Raymond a approfondi son analyse dans « *Homesteading the noosphere* » (Coloniser la noosphère), qui analyse l'économie du don sur lequel repose le développement des logiciels libres [80].

La société du don

« Tout bon travail de développement de logiciel démarre toujours par une démangeaison personnelle chez un développeur »[81]. Pour Eric Raymond, la première motivation pour créer un outil libre est souvent de régler un problème technique posé à un informaticien dans un cas bien précis. L'exemple d'Apache, initié par un webmestre qui avait besoin d'étendre les fonctionnalités de son serveur, est le plus connu. Mais on retrouve la même logique chez tous ceux qui, par exemple, développent des pilotes de périphériques pour que la carte vidéo de leur PC soit compatible avec Linux. Combiné avec ce qu'il faut d'altruisme (on offre à la communauté le fruit de son travail), ce modèle serait presque suffisant pour faire évoluer un logiciel.

Mais le facteur déterminant dans l'accélération des développements est sans doute ce que Raymond appelle « *Ego Gratification* » (la gratification de l'ego). « Pourquoi les peintres amateurs peignent-ils? Parce qu'ils cherchent la reconnaissance de leurs pairs. Pour les développeurs de logiciels libres, c'est pareil », résume John Hall, président de Linux International[82]. De nombreux créateurs de logiciel travaillent d'abord pour l'honneur conquis au sein d'une communauté de développeurs. La gloire acquise par Linus Torvalds pour son système d'exploitation Linux n'a probablement d'égale que celle de M. Bill Gates.

Exploitant ce filon, la société Nintendo organise chaque année au Japon un concours du meilleur créateur de jeux vidéo ; le gagnant voit son jeu distribué sur cartouche mais ne reçoit rien en retour si ce n'est la

80 « Homesteading the noosphere », Eric Raymond, http://www.opensource.org
81 Op. Cit.
82 Propos tenus lors d'une conférence de l'Aful à Bercy Expo, septembre 1998.

gloire d'une diffusion massive de sa création. Cet exemple montre que l'honneur ou la gloire restent des valeurs suffisamment fortes pour qu'un individu, voire une société recherche un succès rapide, massif et mondial, sans en tirer directement de bénéfice financier.

Eric Raymond a systématisé cette idée de richesse immatérielle dans « *Homesteading the noosphere* »[83]. Celui qui donne le fruit de son travail à la collectivité se comporte un peu comme ces guerriers effectuant le *potlach*, cette compétition tribale consistant à offrir à la communauté davantage de dons que ses rivaux. « L'abondance rend les relations de commandement beaucoup plus difficiles à tenir et fait des relations d'échange un jeu presque sans intérêt. Dans les cultures du don, le statut social est déterminé non pas parce que l'on contrôle mais parce que l'on peut donner, écrit Eric Raymond (…). Vu sous cet angle, il est très clair que la société des développeurs *Open Source* est en fait une société du don. A l'intérieur de celle-ci, il n'existe pas de pénurie de produits de première nécessité - espace disque, bande passante, puissance informatique. Le logiciel est librement partagé. Cette abondance créée une situation où la seule mesure valable de réussite face aux autres est la réputation qu'on se fait auprès de ses pairs. » Et donc le code source de qualité que l'on peut donner en contribution à l'effort collectif. Pour les « indigènes » de cette microsociété, il importe avant tout d'être le premier à découvrir, à coloniser (*homesteading*) une zone déterminante de l'espace des produits (exemple : Linux, le premier Unix sous licence GNU) sur un terrain de travail commun (la noosphère).

Le développement coopératif

A l'origine du succès des logiciels *Open Source*, on trouve avant tout un modèle de développement qui permet à tous d'apporter des modifica-

83 Op. Cit. Inventée par le philosophe Theilard de Chardin, la « noosphère » est un nuage immatériel encerclant la terre et composé de tous les inconscients humains émis par les cerveaux droits.

tions. Dans « The Cathedral and the Bazaar »[84], Eric Raymond cite les atouts suivants :

- Comme nous l'avons vu précédemment, le fait que ces produits soient souvent développé en fonction d'un besoin personnel constitue un bon avantage. Si un informaticien en a un besoin urgent, il est rarement le seul.
- Les bons programmeurs savent écrire du code. Les *« grands »* savent comment le réécrire ou le réutiliser. La disponibilité du code source est une ressource sans égale pour ces développeurs.
- Il faut jeter. Travailler en groupe sur un code source commun permet de sélectionner et de raffiner longuement les modifications brutes proposées par chacun.
- L'ouverture du code permet de faire de n'importe quel utilisateur un programmeur potentiel.
- Les mises à jour très régulières - car non régies par des impératifs commerciaux - permettent d'intégrer rapidement les réactions des utilisateurs.
- Ce qui a pour conséquence : une base de développeurs et de bêta testeurs beaucoup plus large que pour un projet commercial. Et un développement/déboguage accéléré.

C'est la grande règle retenue par Eric S. Raymond : les communautés du logiciel libre permettent un développement et un déboguage parallèles. Des centaines de personnes développent, testent, améliorent et font connaître leur réaction en même temps.

Mais ce modèle de développement collectif, presque en temps réel, n'aurait jamais vu le jour sans la diffusion de l'Internet. Les équipes de développeurs sont en effet souvent disséminées à travers le monde. Le noyau de Linux a été conçu par un groupe de 200 contributeurs, issus aussi bien des continents européen qu'américain ou asiatique. Ces projets

84 Op. Cit.

reposent d'ailleurs sur toutes les ressources possibles du réseau mondial : listes de diffusion, groupes de discussions, sites Web…

L'avenir d'une idée

Un projet de logiciel libre est souvent coordonné au départ par un seul individu. C'est le cas de Sendmail avec Eric Allman ou de Linus Torvalds avec Linux. Eric S. Raymond parle alors de « dictateur bénévole ». Mais lorsque le produit commence à avoir du succès, une communauté d'utilisateurs prend souvent le relais du ou des créateurs du logiciel pour assurer sa maintenance et son développement. Cette communauté comprend deux groupes : les utilisateurs passifs, qui utilisent le logiciel et signalent les bogues, et les utilisateurs actifs, qui utilisent le logiciel et proposent des corrections ou des améliorations.

Lorsqu'un membre clef de la communauté cesse de participer au développement du logiciel, un autre membre prend le relais. Lorsqu'un membre important de la communauté entend commercialiser le code qu'il a créé, les autres membres font corps et poursuivent seuls le développement du logiciel libre. Les contrats de licence des logiciels libres permettent en effet à n'importe quel individu de créer un nouveau produit à partir d'un code source existant et sans payer de droits ; on parle alors de phénomène de divergence.

Les communautés d'utilisateurs actifs tendent toutefois à devenir moins spontanéistes, tout en restant coopératives. Les améliorations de la version standard d'Apache sont votées par un collège de membres (en l'occurrence, l'Apache Group) mais peuvent être proposées par tous. Même chose pour l'interface graphique KDE, dont les propositions de modifications, sont validées par un petit groupe de développeurs. Les différentes versions font d'ailleurs l'objet d'une certification par ce même groupe.

VRAI OU FAUX : le monde du logiciel libre est dominé par des principes d'entraide

L'entraide est très forte sur les forums. Le magazine Infoworld a décerné à la communauté Linux, la médaille du meilleur support utilisateur pour 1997. Mais les sociétés commerciales qui travaillent dans ce domaine sont soumises à des impératifs de rentabilité - et ne sont donc pas plus généreuses que les autres. On constate une facturation parfois très élevée de l'installation de Linux. « Une solution libre n'est pas forcément moins chère », insiste, par exemple, Lucien Petit, directeur commercial d'Alcôve, SSII spécialisée dans les solutions libres.

Côté développeur, créer ou améliorer un logiciel libre est un bon moyen de se faire de la publicité. Quant aux sociétés faisant des développements autour de Linux, elles développent souvent des produits propriétaires, à code source et formats fermés. Leurs clients sont donc captifs et ne profitent donc pas de l'ouverture dont a tiré profit leur fournisseur.

Les limites du modèle coopératif

La prise en charge du développement par des communautés d'utilisateurs provoque parfois les foudres de certains directeurs informatiques, qui les accusent d'être des groupes de développeurs « fantômes » susceptibles de disparaître du jour au lendemain. Mais, d'une part, ces communautés peuvent être très structurées, à l'image de l'organisation collégiale de l'Apache Group. D'autre part, la disponibilité du code et le droit universel de le modifier garantit une reprise du développement par n'importe quel utilisateur. Contrairement au logiciel propriétaire dont l'évolution peut cesser d'être assurée par manque de rentabilité du produit ou par défaillance de la société qui en a la propriété, un logiciel libre ne cesse d'être développé que faute d'utilisateurs.

Autre limite du modèle de développement coopératif : il n'existe aucune garantie de contrôle qualité sur le développement du produit, dans la mesure où n'importe qui peut se poser en mainteneur, c'est-à-dire en coordinateur du développement d'un produit. Mais les participants tendent à se regrouper autour des versions les plus fiables et les plus

complètes.

Certains analystes accusent aussi ce modèle de recourir à des développeurs concevant un produit certes communautaire, mais… pendant leurs heures de travail. Ce qui représenterait en réalité un coût caché pour la collectivité. « Un logiciel gratuit qui évolue grâce aux contributions de milliers d'informaticiens, d'universitaires et de scientifiques du monde entier ne représente-t-il pas un coût pour quelqu'un? »; s'interroge par exemple Jean-Marie Sepulchre, maître de conférence associé à l'Université d'Orléans dans une tribune parue dans « 01 Informatique »[85]. « On peut estimer que des dépenses d'enseignement et de recherche financées -quand elles sont publiques- par le contribuable planétaire sont une contrepartie de cette gratuité. »

Certains produits ont certes été développés grâce à des bourses d'études (c'est l'exemple du premier noyau de Linux). Mais on peut aussi retourner l'argument : ne pas exploiter ces travaux sous forme de logiciels libres reviendrait à demander au contribuable de payer à nouveau pour ce qu'il a déjà financé.

Cependant, la plupart des logiciels libres ont été créés dans le cadre d'organisations privées autofinancées (exemple : les logiciels GNU développés par la FSF) ou par des informaticiens ayant eu besoin de développements spécifiques pour garantir le bon fonctionnement de l'informatique de leur entreprise. C'est le cas d'Apache, dont le développement a été assuré par des professionnels ayant besoin d'un serveur Web plus sophistiqué que l'offre disponible. Aujourd'hui, une bonne partie du développement des logiciels libres est assuré via des contributions volontaires (financement de salaires de développeurs en interne ou de développements spécifiques) de sociétés vivant de logiciels libres ou les utilisant. Mais la majorité des développeurs participent à la création des logiciels libres… en dehors de leurs heures de travail.

85 « 01 Informatique » du 11 décembre 1998.

Le rôle des associations

Il ne suffit pas de « savoir faire » des logiciels libres, par l'agrégation de compétences. Il faut aussi « faire savoir », c'est-à-dire disposer de l'équivalent d'un service marketing pour faire connaître le produit. Ce rôle est imparti aux médias - notamment sur le Web - et aux organisations internationales (FSF, Linux International, OSI, etc.), nationales (Aful, April, etc.) ou régionales qui font la promotion de l'offre en logiciels libres. Ces bénévoles compensent par l'enthousiasme l'absence de budget communication ou de forces de ventes rémunérées. Ceci dit, ces associations font souvent alliance avec des acteurs de la - toute fraîche - industrie du logiciel libre.

A l'échelle internationale

Le grande ancêtre des organisations de promotion des logiciels libres est la Free Software Foundation (FSF), fondée en 1984 par Richard Stallman, alors chercheur au MIT (voir Chapitre 1). Le but initial était de créer et de diffuser des logiciels libres - notamment un Unix - dans le cadre du projet GNU (« GNU's Not Unix »). Aujourd'hui, la Fondation vit avant tout de la vente des copies sur CD-ROM et des manuels d'utilisation des outils qu'elle a conçu (notamment l'éditeur de texte Emacs et le compilateur gcc).

Plus tard, le besoin de promouvoir Linux a donné naissance à Linux International, présidé par John Hall, senior manager chez Digital-Compaq. Cette organisation à but non lucratif, basée à Amherst aux Etats-Unis, est chargée de diffuser l'information concernant l'Unix libre, de participer à des foires commerciales pour promouvoir cette plate-forme et de faire tester les dernières versions du produit par les programmeurs.

A mesure que Linux s'est intégré à l'économie traditionnelle, les sociétés spécialisées dans la commercialisation de Linux (SuSE, Caldera, Red Hat, LinuxPPC Inc.,…) ont rejoint l'organisation. Surtout, la plupart des

acteurs intéressés par l'intégration de cette plate-forme à leur offre sont devenus membres (SCO, Compaq, Silicon Graphics, Sun, Informix, etc.). La plupart d'entre eux participent d'ailleurs au financement des développements et des tests sur l'OS libre.

De Bluetooth pour la technologie de communications à ondes courtes à l'Object Management Group (OMG) pour le modèle objet Corba, en passant par Tron pour l'embarqué, les groupements d'industriels finançant le développement de spécifications communes sont courants. Avec Linux International, on pourrait donc considérer que ce type de consortium est étendu aussi à des logiciels communs. Avec une différence de taille : les industriels participent au développement en tant qu'utilisateurs d'une technologie, au même titre que les autres utilisateurs. L'intérêt des grands groupes ne prévaut pas sur celui des autres membres.

Avec la diffusion du modèle *Open Source* dans l'industrie, certains défenseurs du logiciel libre ont décidé de créer l'*Open Source Initiative* (OSI). Présidée par Eric S. Raymond, elle a pour objectif de défendre le label *Open Source* contre d'éventuelles récupérations. Pour ce faire, l'OSI a déposé un nom de marque pour **«** *Open Source* **»** grâce auquel elle compte certifier les projets et les logiciels répondant effectivement au modèle de développement à code source ouvert.

Les centrales d'informations

La faiblesse de couverture médiatique comme de la publicité sur les logiciels libres a longtemps été compensée par une forte présence sur le Web. Des sites comme Slashdot.com sont ainsi devenus de véritables portails d'information continue sur le monde de l'*Open Source*. Les sites spécialisés dans les produits (BSD, Linux, Gimp, etc.) remplissent un rôle identique dans leurs domaines, tout en offrant en outre la possibilité de télécharger les outils dont ils traitent et des modes d'emploi (les *How-To*) pour s'en servir.

On retrouve des organes d'information similaires en France, la plupart centrés sur Linux. La plus connue est la revue de presse « les Nouvelles Neuves de Linux » du Linux-Center, reprise sur le site de l'Aful, complétée par celle de « Linux-Business.com » de Jacques Le Marois, et plus récemment, par le généraliste « Linufr.org ».

La presse papier suit se mouvement dans la plupart des pays industrialisés : « Linux Journal » aux Etats-Unis, « Dream » et « Linux Magazine » en France, « Linux Japan » au Japon, pas moins de trois magazines en Allemagne, des magazine en Espagne, dans les pays du Nord de l'Europe, etc.

Les clubs d'utilisateurs

Les *newsgroups* et les mailing-lists servent de relais mondiaux aux utilisateurs de logiciels libres ayant besoin d'un conseil, voire d'un *patch* ou d'une information. Il ne sont d'ailleurs pas les seuls, les utilisateurs de Mac et de NT ont aussi les leurs. Mais les utilisateurs ne disposant pas du code source, il leur faudra attendre les rustines fournies par l'éditeur de leur OS.

Ces lieux d'échanges font d'ailleurs l'objet de localisations : un forum de discussions « fr.comp.os.linux » et une liste « libre » (pour les applications *Open Source*), existent depuis longtemps pour les utilisateurs francophones.

Les associations de promotion en France

Le phénomène associatif dans le monde du logiciel libre est très ancien. Déjà en septembre 1988, les utilisateurs de TeX s'étaient regroupés autour de Jacques Beigbeder et de Bernard Gaulle au sein de l'association Gutenberg[86]. L'association conçoit et diffuse des distributions de TeX et LaTeX adaptées au français (Unix, MacOS et Windows). Elle compte au-

86 http://www.gutenberg.eu.org

joud'hui près de 800 membres.

Quelques années après, l'association informelle Freenix se lança dans l'organisation régulière de repas amicaux entre programmeurs et utilisateurs d'Unix libres. La plupart des membres s'étaient connus à travers les forums de discussions de BBS associatifs qui préfiguraient ce que sont aujourd'hui les prestataires privés d'accès à Internet et les *newsgroup*. René Cougnenc, personnage fédérateur de Freenix, fut aussi l'un des premiers en France à parler et à diffuser Linux avant de se donner la mort. Les membres de Freenix sont pour la plupart des professionnels des réseaux que l'on retrouve un peu partout dans les instances d'Internet. Ils gèrent le domaine gratuit eu.org (utilisé par exemple par Gutenberg) et continuent à organiser des repas hebdomadaires au Finnegan's Wake, tous les mercredi soir pour boire quelques bières.

Mais ni Gutenberg ni Freenix n'affichaient d'objectif prosélyte.

En 1996, des informaticiens, étudiants et anciens étudiants de l'université Paris 8 ont créé l'April (association pour la promotion et la recherche en informatique libre), dirigée par Frédéric Couchet, Architecte système chez Cap Gemini. Les membres fondateurs d'April ont, lorsqu'ils étaient étudiants à Paris-Vincennes, « introduit à grande échelle les outils GNU dans l'université aux débuts des années 90. Aujourd'hui, les logiciels libres sont omniprésents dans le centre de calcul de l'université (GNU/Linux, Free/Net BSD, ...) », précise Frédéric Couchet.

Proche de la FSF américaine et des idées de son fondateur Richard Stallman, « elle a pour objectif de sensibiliser les entreprises, les administrations et les particuliers sur les risques des solutions propriétaires et fermées, et de les informer des potentialités offertes par les logiciels libres et les solutions basées sur des standards ouverts ». L'April organise régulièrement des séminaires auxquels participe parfois Richard Stallman lui-même. L'April soutient des projets locaux en province et est implantée dans les SSII (Cap Gemini, Alcôve) et l'Education nationale.

Plus récemment, en avril 1998, s'est créée l'Aful (Association francophone des utilisateurs de Linux et des logiciels libres), dirigée par Stéfane Fermigier, maître de conférences à Paris VII et Bernard Lang, directeur de recherche à l'Inria. L'Aful a pour objectif principal la promotion des « logiciels libres et en particulier les systèmes d'exploitation libres, principalement ceux basés sur les normes POSIX ou dérivées (c'est-à-dire de type Unix), dont le plus connu est le système Linux muni de l'environnement GNU ». Néanmoins, l'Aful ne défend le point de vue selon lequel tout logiciel doit être libre. L'Aful milite aussi pour « que les normes de communications soient publiquement documentées et librement uilisables», indique Bernard Lang, secrétaire de l'association.

L'Aful rassemble des personnalités très variées : des débutants et des utilisateurs confirmés, des utilisateurs et des professionnels, des étudiants, des enseignants, des chercheurs et des cadres dirigeants, etc. L'Aful est un partenaire officiel du Ministère de l'éducation nationale dans le cadre du plan de mise en réseau des écoles, lycées et collèges. La plupart des entreprises du logiciel libres sont membres de l'Aful (Pick Systems, Netscape, SQL Ingéniérie, Alcôve, Mandala International, Atrid, etc.).

Linux de bonne volonté

Parce qu'ils sont conçus par les utilisateurs et teintés d'une forte empreinte idéologique, les logiciels libres suscitent la création de nombreuses associations de terrain, majoritairement créés autour de Linux.

Les Lug (*Linux User Group*) régionaux organisent ainsi des *install party*, sortes de fêtes où on peut venir se faire installer l'OS libre par des utilisateurs plus expérimentés, mais mènent aussi des actions de promotion ou des micro-projets mettant en œuvre des solutions libres. Parmi les grands ancêtres, Freenix, l'association parisienne des utilisateurs d'Unix libres, la GUILDE rhône-alpine et le CULTe toulousain. Durant l'année 1998, el-

les se sont multipliées dans l'Hexagone, de la Provence (Plug) à la Bretagne, avec le Gulliver en Ile-et-Villaine et le Finix en Finistère, en passant par l'Alsace (Flammekueche Connection).

Le rôle du lobbying

Au niveau mondial, la promotion de Linux auprès des directions des grands constructeurs (Compaq, HP, etc.) par des employés de ces sociétés s'est révélée déterminante pour l'adoption de ces plates-formes. En France, les associations nationales ou régionales ont toutes un rôle de lobbying soit auprès des entreprises, soit auprès des administrations. « Linux possède les plus fortes ressources mondiales en matière de recherche et développement, mais n'a pas de force de ventes. Le rôle de l'Aful est de jouer le rôle du service marketing », résume Bernard Lang, secrétaire de cette association. Alors que parallèlement, April s'active dans les SSII, l'Aful s'est construite des relais efficaces dans les ministères.

En France, le *lobbying* au sommet des gouvernements se révèle en effet nécessaire à la diffusion massive d'un logiciel, comme le prouve la stratégie des grands éditeurs. Lorsque le CEO de Microsoft Bill Gates est venu à Paris en 1996 et 1997, il a été reçu par le président Jacques Chirac, le Premier ministre Lionel Jospin et le ministre de l'Economie et des Finances, Dominique Strauss-Kahn. Microsoft n'hésite pas non plus à inviter à ses frais aux Etats-Unis les responsables gouvernementaux pour leur montrer à quel point ses logiciels sont nécessaires pour consolider la compétitivité de leurs nations.

Cette pratique est commune à la plupart des sociétés. Le groupe de services Arthur Andersen a, par exemple, invité aux Etats-Unis les spécialistes des technologies de l'information de plusieurs cabinets des ministères pour montrer à quel point un environnement tel que Lotus Notes pouvait être utile pour organiser une administration. En février 1998, l'Education nationale a d'ailleurs adopté Notes comme plate-forme de

Logiciel libre et infopauvres

Dans le cas des structures disposant d'un faible budget (associations, écoles ou collectivités locales), l'usage de logiciels libres permet une baisse du coût d'informatisation.

- Ils sont souvent gratuits et toujours meilleur marché que leurs équivalents commerciaux. Pour les applications disponibles uniquement sous licence propriétaire, le prix est le plus souvent inférieur pour les plates-formes libres.
- Ils permettent de recycler un matériel obsolète (486 Intel ou Motorola 68030) et de le combiner avec du matériel récent (formats ouverts et compatibles avec des versions anciennes des logiciels).

Dans le tiers-monde, il offre une garantie de développement durable (*sustainable development*) : la plupart des pays pauvres ne peuvent payer les licences des éditeurs commerciaux. Cela met en danger la compétitivité de leurs entreprises, qui soient ne s'équipent pas, soient ne paient pas les licences de leurs produits et s'exposent donc à des poursuites qui menacent leur survie. Exemple : une agence de publicité brésilienne, Artplan, poursuivie par la *Business Software Alliance* (BSA) qui lutte contre le piratage pour le compte des grands noms du logiciel, a été condamnée à verser une amende de 65 millions de dollars pour l'utilisation frauduleuse de 382 programmes, soit 170 000 dollars le logiciel.

Il n'est donc pas étonnant que Linux soit très répandu dans les pays qui entrent tout juste dans la société de l'information (Afrique, Russie,…) ou sont privés de transferts de technologies du fait des embargos (Iran, Yougoslavie, etc.). Même chose pour les *freeware* et les *shareware* : pendant la guerre, à Sarajevo, le *shareware* Zerberus a servi de logiciel de messagerie pour le nœud local du BBS Zamir, qui reliait la ville au reste de l'ex-Yougoslavie et à l'Internet. Faute de téléphone ou de courrier postal, ce réseau était le seul moyen de communiquer avec l'extérieur dans la ville assiégée.

Zerberus était à l'origine un *freeware* créé pour l'un des premiers grands réseaux alternatifs allemands, Fidonet.

choix pour l'informatisation de son administration centrale.

Mais l'Aful a, elle-aussi, des relais au sein des ministères, ce qui a permis en moins d'un an un retournement significatif des positions gouvernementales concernant les logiciels libres.

De Gates à Linux : histoire d'un basculement

Lors du discours d'Hourtin en août 1997, le Premier ministre Lionel Jospin avait donné le coup d'envoi à une politique volontariste d'informatisation du service public. Un peu plus tard, la visite de Bill Gates à Paris et l'accueil enthousiaste que lui avait réservé le ministre des Finances Dominique Strauss-Kahn, pouvait laisser croire que les solutions Windows seraient favorisées. D'autant que le ministre souhaitait s'appuyer sur cette rencontre pour obtenir l'implantation d'un laboratoire de recherche Microsoft en France, ce qui aurait permis de reproduire la réussite du centre d'IBM à la Gaude.

Pourtant, parallèlement, un conseiller technique de Claude Allègre avait chargé officieusement Bernard Lang, chercheur à l'Inria, d'étudier la viabilité, en tant qu'alternative, des logiciels libres dans l'enseignement. De son côté, Jean-Paul Baquiast, contrôleur d'Etat et président d'Admiroutes, avait recommandé l'utilisation de ce type de solutions dans son rapport sur l'informatisation de l'administration rédigé pour le Ministre de la fonction publique, Emile Zucarelli. Il démontrait notamment que les logiciels libres permettaient de faire baisser les coûts, au profit notamment de la création d'emplois et du contribuable.

Depuis, Jean-Paul Baquiast est devenu président du conseil d'administration de l'Aful et Bernard Lang son secrétaire. La mise en commun des réseaux de pouvoirs croisés d'universitaires et de hauts-fonctionnaires a permis de faire du *lobbying* dans les ministères et aussi dans l'industrie, en lieu et place d'une organisation commerciale. Ce qui lui a valu notamment de signer en octobre un accord-cadre avec le ministère de l'Education nationale pour aider au déploiement de solutions libres dans l'enseignement.

Enfin et à titre d'exemple, le *lobbying* de l'Aful n'est peut-être pas étranger à l'évolution du point de vue de Dominique Strauss-Kahn qui, après avoir salué, à l'automne 1997, l'innovation apportée par Microsoft

à l'informatique, recommandait l'année suivante de veiller « à ce qu'une offre pluraliste soit possible (…). N'oublions pas la révolution apportée par Internet et qu'illustre l'importance croissante des logiciels libres comme Linux. »

Ce basculement n'aurait pas non plus été possible sans l'engagement de médias, en particulier le quotidien « Libération[87] » et « Le Monde Diplomatique[88] », qui n'ont eu cesse de défendre une conception pluraliste de l'informatique en accordant une place importante dans leurs colonnes à des points de vues iconoclastes.

Les limites de l'informatique citoyenne

La prise en charge de la promotion des logiciels libres par des organisations à but non lucratif a parfois suscité la méfiance de certaines entreprises, qui voyaient dans ces groupes des structures à caractère plus éthique ou idéologique que technique. La validation d'une partie de l'offre libre (notamment Linux) par l'industrie leur a permis de faire la part entre la composante philosophique des logiciels libres et leur utilité pratique.

L'intégration croissante de l'industrie et des organisations de promotion des logiciels libres comporte des risques de dérive non négligeables. Le financement des organisations et des projets structurés autour des logiciels libres (ex : Linux International ou l'OS embarqué eCos) par des industriels pourrait à terme infléchir les développements de ces produits en fonction des besoins des industriels et non de ceux des utilisateurs. Cette inflexion reste peu probable du fait de la disponibilité du code de source, qui permet aux utilisateurs de continuer à contrôler l'évolution d'un produit.

---

87 «Linux, Mini OS contre maxi-exploitation», par Jean-Claude Guédon et Bernard Lang, Libération du 7 novembre 1997.

88 « Des logiciels libres à la disposition de tous », Bernard Lang, Le Monde diplomatique, janvier 1998. http://www.aful.org/publi/articles/diplo.html

EN BREF – Le développement des logiciels libres est d'abord défi tribal « pour la gloire » avant d'être pris en charge par une micro-société de développeurs-utilisateurs lorsqu'il devient un produit digne de ce nom. La participation à l'évolution du produit peut parfois déboucher sur une logique commerciale. Mais le cadre juridique permet alors aux développeurs à but non lucratif de poursuivre l'enrichissement du produit.

En l'absence de service marketing, la promotion des logiciels libres est assurée par des associations, de l'échelle internationale au niveau régional. En France, l'Aful et l'April sont les deux associations nationales les plus actives.

8

Des logiciels libres pour réguler le marché

La politique doit-elle encore se mêler des affaires de l'industrie informatique et soutenir un nouveau plan calcul ? « Trois échecs n'ont-ils pas déjà suffit ? » Un tel constat, souvent entendu dans la bouche de hauts fonctionnaires français, conduit à douter des bienfaits de l'action publique dans le domaine des technologies de l'information. Et les mésaventures de Bull, des Micral ou des TO7 semblent le confirmer.

Mais, ce qui est bon sur un marché parfait avec des rendements décroissants ne l'est pas forcément dans une industrie du logiciel dont les caractéristiques - présences d'oligopoles, rendements croissants et coûts marginaux nuls - conduisent parfois à des tarifications abusives et à des situations contraires à l'intérêt général. En effet, le cadre juridique du marché du logiciel facilite l'exécution de stratégies de domination et de prédation particulièrement efficaces, fondées sur l'appropriation des nor-

mes de communication au profit d'un seul acteur[89].

Les logiciels libres permettent de corriger ces dérives, comme l'illustre le serveur Web Apache. En effet, lorsque le standard du marché est un logiciel libre, les stratégies d'appropriation des normes de communication deviennent largement inopérantes. Les logiciels libres agissent alors comme régulateurs du marché et favorisent l'émergence d'une offre diversifiée de logiciels commerciaux compatibles.

Aussi, en affirmant dans un discours précurseur[90], que « …pour éviter tout monopole… n'oublions pas l'importances des logiciels libres comme Linux », Dominique Strauss-Kahn annonce ce que pourrait être une politique de la concurrence dans le domaine des technologies de l'information et de la communication : utiliser des logiciels libres, développer des logiciels libres et favoriser les actions concertées autour des logiciels libres.

Les dérives de l'industrie du logiciel

Le choix d'un logiciel par rapport à un autre s'effectue souvent en fonction de sa capacité à communiquer avec autrui et rarement en raison de son prix ou de ses qualités intrinsèques. Lorsqu'une majorité d'acteurs économiques tient ce raisonnement et que le logiciel le plus utilisé communique mal avec les autres, ses parts de marché ne cessent d'augmenter jusqu'à supprimer toute concurrence en dehors de niches. C'est ce que l'on appelle le phénomène de standard du marché. Ce phénomène, bien connu en économie des réseaux, a contribué à la marginalisation du système MacOS, qui pouvait lire les disquettes DOS, et à la victoire du système DOS, qui était incapable de lire les disquettes MacOS. DOS coûtait

89 Pour la cas particulier de Microsoft, voir par exemple « Le hold-up planétaire : la face cachée de Microsoft ». Roberto Di Cosmo, Dominique Nora. Calmann-Lévy.

90 Discours du Ministre de l'Economie, des Finances et de l'Industrie. Hourtin, 28/8/98.

certes plus cher à mettre en œuvre[91] mais était majoritaire.

Il arrive souvent qu'un producteur dominant choisisse de ne pas respecter les normes communes. On était par exemple frappé de constater qu'en 1997, des logiciels de messagerie électronique produits par des sociétés aussi prestigieuses que Microsoft ou IBM ne respectaient pas pleinement les normes de communication de l'Internet alors que des micro-sociétés telles que Stalker[92] les suivaient parfaitement et en temps réel avec des produits beaucoup moins coûteux. Aujourd'hui encore, de nombreux produits Microsoft dont le traitement de texte Word ne suivent pas pleinement les normes de l'Internet pour la publication sur le Web ou la gestion des caractères accentués français.

La stratégie du Winner Take All

Le phénomène de standard du marché s'est reproduit dans la plupart des secteurs de l'informatique où la stratégie d'un acteur dominant a consisté à brider les capacités de communication de ses propres logiciels avec ceux de la concurrence. Que ce soit dans la bureautique (Microsoft Office), dans l'édition (Quark Xpress) ou dans l'industrie textile (Gerber), dès qu'un éditeur en position dominante détériore les capacités de communication de ses logiciels, soit par une évolution permanente de normes de communication tenues secrètes, soit par des brevets sur des normes de communication publiées, les concurrents commencent par adopter la même stratégie pour conserver un marché d'utilisateurs captifs avant d'être marginalisés par l'acteur dominant en raison du phénomène de standard du marché.

C'est la stratégie du *Winner Take All* (« le gagnant prend tout »). Elle justifie les investissements marketing considérables des *start up* pour acheter une part du marché ou une part des esprits (*mind share*) dans une logique où l'acteur dominant finit par emporter tout le marché et jouit

91 http://www.mackido.com/Myths/cost.html
92 http://www.stalker.com

d'une rentabilité exceptionnelle alors que les autres acteurs sont condamnés à disparaître.

Mais la stratégie de *Winner Take All* est contraire à l'intérêt des utilisateurs car elle tend à favoriser les stratégies de fermeture et d'incompatibilité des logiciels. La profession du textile souffre par exemple depuis de nombreuses années des incompatibilités chroniques entre les systèmes de conception Lectra et Gerber qui se disputent le marché. Ces incompatibilités obligent parfois une société de confection utilisant les systèmes Lectra à reprendre à la main l'ensemble des données d'un modèle de vêtement conçu avec un système Gerber ce qui entraîne pour toute une profession d'importants surcoûts. Et, par crainte de perdre leur marché captif, ni Lectra ni Gerber n'envisagent de faire le premier pas vers une véritable ouverture de leurs logiciels.

La loi de la plate-forme

Les phénomènes de standard du marché et la stratégie de *Winner Take All* se généralisent rapidement, d'un logiciel isolé à toute une plate-forme. Comme le note Bernard Lang[93], « dès lors qu'une plate-forme domine le marché, tout créateur de logiciels commerciaux s'adaptera préférentiellement à la plate-forme dominante, pour des raisons évidentes de rentabilité. C'est ainsi le cas des fournisseurs de périphériques pour PC ou des éditeurs de jeux vidéo qui fournissent toujours leurs produits pour les plates-formes logicielles Microsoft et plus rarement pour les autres (MacOS, OS/2, Linux, etc.)».

Ce phénomène est renforcé par la fermeture des standards propriétaires du marché qui rend difficile, faute d'information suffisante ou en raison de protections juridiques, le développement par des tiers de logiciels équivalents et compatibles sur les plates-formes ainsi négligées. Il est par

93 « Ressources Libres et Indépendance Technologique dans les Secteurs de l'Information ». Bernard Lang. Colloque Inforoutes et Technologies de l'Information, Forum Initiatives 97, Hanoi, 25-26 octobre 1997.

exemple très difficile d'exploiter sur une plate-forme Linux une encyclopédie sur CD-ROM prévue pour Windows alors que cela ne pose en principe aucun problème technique.

L'arme des brevets

Certains éditeurs en position dominante sont parfois gênés lorsqu'une *start up* met sur le marché un nouveau logiciel susceptible de rencontrer un grand succès et de menacer ses parts de marché ou, dans le cas d'un logiciel très innovant, ses parts d'esprit (*mind share*) et sa notoriété. En rachetant ces *start up*, les grands éditeurs qui se sentent menacés peuvent conserver leur position tout en faisant la fortune des créateurs d'entreprise. Mais il n'est pas toujours nécessaire de racheter une *start up* pour l'éliminer. Plus que dans tout autre industrie, les brevets sur les logiciels (illégaux en Europe, voir chapitre 5) permettent de rayer du marché une entreprise en l'attaquant pour contrefaçon.

Le bureau américain des brevets accepte en effet environ 20 000 brevets chaque année (35 000 au Japon) sur des procédés liés aux logiciels qui sont le plus souvent des applications immédiates de formules mathématiques, de principes d'organisation ou de procédés d'interface élémentaires. La fonction mathématique « ou exclusif » a par exemple été brevetée pour son application au procédé d'inversion des couleurs sur un écran comme celui qui est utilisé pour faire clignoter un curseur sur un écran. Or, il s'agit d'un procédé connu et tellement élémentaire que personne n'avait osé déposer un brevet auparavant. Il s'agit aussi du procédé le plus naturel et le plus efficace ce qui interdit de s'en passer. De nombreuses sociétés ont alors été attaquées pour contrefaçon parce qu'elles utilisaient la fonction « ou exclusif » dans des applications graphiques sans savoir qu'un brevet existait et ont dû accepter de verser des *royalties* au détenteur du brevet pour éviter un coûteux procès.

Mais rien n'oblige le détenteur d'un brevet à accorder un droit d'usage

de son invention. Le brevet constitue donc une arme potentielle pour permettre à son détenteur d'interdire la mise sur le marché d'un nouveau produit. C'est d'ailleurs une pratique courante dans des industries comme la chimie ou la pharmacie. Or l'industrie du logiciel est une industrie encore jeune dans laquelle les véritables inventions sont noyées dans un flot d'applications immédiates de formules mathématiques élémentaires ou de principes d'organisation souvent suggérées par les utilisateurs eux-mêmes. Les brevets sur les logiciels sont donc peu coûteux en termes de recherche et de développement. Et comme le système est noyé, les recherches d'antériorité ne sont pas effectuées sérieusement ce qui conduit au dépôt de nombreux brevets sans valeur.

De grandes multinationales du logiciel comme IBM, Apple, Sony, etc. qui disposent de moyens importants pour financer les procédures de dépôt, ont compris tout l'intérêt qu'elles avaient à se constituer des collections de brevets logiciels sur des procédés suffisamment élémentaires et fondamentaux pour pouvoir bloquer tout concurrent gênant et se protéger en cas d'attaque. La plupart des plaintes pour contrefaçon se résolvent par un échange de brevets. Néanmoins, en cas de poursuites juridiques sérieuses, seuls les petits acteurs sont véritablement menacés car la plupart des brevets n'ont pas de valeur. Ainsi, une *start up* française qui avait émigré aux Etats-Unis a du cesser ses activités après avoir été attaqué pour contrefaçon car elle n'avait pas les moyens de financer sa défense juridique. La société Adobe, attaquée ensuite pour le même brevet, est parvenu lors du procès à le faire annuler. Faute de moyens financiers suffisants, la *start up* française a donc été éliminée par un brevet sans valeur.

Echec aux stratégies de *Winner Take All* : l'exemple d'Apache

Le marché des serveurs Web est dominé par le logiciel Apache, un logiciel libre qu'IBM a intégré à son offre de services autour du commerce électronique et a adapté à ses serveurs AS/400. Plus de 50% des serveurs Web du monde entier utilisent Apache en raison de ses performances, de sa stabilité, de sa haute flexibilité, de sa gratuité et de sa conformité aux normes de l'Internet. Apache a été développé par un groupe d'utilisateurs ayant besoin d'un serveur Web à hautes performances. Au lieu d'essayer de vendre leur logiciel et de se retrouver confrontés à la puissante force de marketing des grands éditeurs, ces utilisateurs ont choisi de mettre Apache à la disposition de tous en espérant pouvoir bénéficier en retour des améliorations apportées par d'autres.

C'est ainsi qu'Apache peut désormais être complété par divers modules d'optimisation et de chiffrement dont certains, développés dans des pays européens libéraux, peuvent être légalement exportés dans le reste du monde. Apache a conquis rapidement le marché du site Web car il était à l'époque le seul à offrir un niveau suffisant de fiabilité, de performance et de prix pour pouvoir vendre à moins de 2 dollars l'hébergement une vitrine complète de commerce électronique.

Mais, contrairement aux éditeurs de logiciels commerciaux, les auteurs des logiciels libres n'ont aucun intérêt à fermer les normes dans une logique de stratégie de *Winner Take All*. Au contraire, pour que de nombreux utilisateurs puissent collaborer au développement d'un logiciel sans se connaître, il importe que chacun puisse travailler individuellement dans le respect de normes compréhensibles et largement publiées. Par ailleurs, pour que puisse être garantie la libre diffusion des logiciels libres, ceux-ci doivent s'appuyer sur des normes publiques clairement documentées et libres de droits.

Aussi, des éditeurs de logiciels commerciaux sont en mesure de pro-

poser des produits concurrents compatibles avec le standard libre du marché. Dans le domaine des serveurs Web, on trouve de nombreux produits commerciaux édités par de grands éditeurs comme Netscape ou Microsoft et par des *start up* comme Star Nine (Webstar pour MacOS) ou Roxen[94] (Challenger, logiciel suédois primé au Comdex 1998). Tous ces produits utilisent les mêmes normes que celles d'Apache et sont rentables dès qu'ils couvrent un besoin non satisfait par Apache (interface utilisateur, fonctions avancées, technologie innovante, application ciblée, etc.). Réciproquement, les normes propriétaires comme l'ASP que Microsoft tente d'imposer à travers le serveur Web de Windows NT, sont parfois intégrées à Apache pour devenir *de facto* des normes publiques.

Pour une politique de la concurrence

Action 1 : *utiliser les logiciels libres*

L'exemple d'Apache montre qu'un logiciel libre devenu le standard du marché permet de prémunir les utilisateurs contre les stratégies de *Winner Take All* d'un éditeur tout en préservant la concurrence et la diversité. Or, il existe de nombreux autres logiciels libres qui, d'un point de vue économique, technique et industriel, sont équivalents aux meilleurs logiciels commerciaux : Linux dans le domaine des systèmes d'exploitation, eCos dans le domaine de l'informatique industrielle et embarquée, les outils GNU dans le domaine du développement, Gimp dans le domaine de la retouche photographique etc.

Il suffirait qu'un nombre important d'utilisateurs choisisse ces logiciels plutôt que des équivalents propriétaires pour que, mécaniquement, les stratégies de *Winner Take All* soient désactivées sur un nombre croissant

94 Roxen est d'ailleurs un logiciel libre car son éditeur suédois souhaite une adoption rapide et espère vendre aux utilisateurs des modules d'extensions propriétaires et du service.

de segments du marché. Par leur influence naturelle, l'administration et les grands groupes sont probablement les mieux placés pour remplir ce rôle en devenant les prescripteurs volontaires de logiciels libres.

Cette mission peut être accomplie par la tactique du « faire et faire savoir ». « Faire », c'est utiliser sur le terrain des logiciels libres. « Faire savoir » c'est utiliser les réseaux d'information pour expliquer les qualités, et parfois les défauts, des logiciels libres.

La DRIRE[95] Lorraine utilise par exemple un *firewall* à base de Linux pour accéder à Internet. Ce produit a été développé localement par une société de service messine, Locabureau[96], qui, grâce à la disponibilité de nombreux composants sous formes de logiciel libre, a pu concevoir un produit innovant sans avoir à effectuer un important investissement de départ. Cette utilisation peut avoir valeur exemplaire et permettre la diffusion des logiciels libres au sein des administrations ou des PME qui, à l'occasion d'une demande d'aide, peuvent se renseigner sur les choix techniques de la DRIRE pour disposer d'un accès sécurisé à Internet.

Un autre exemple est celui de la Communauté Urbaine de Lille qui, après être parvenu à déployer un Intranet de grande taille au moyen de logiciels libres, en a fait une présentation très convaincante à l'hôtel Matignon devant des hauts responsables[97] lors d'un séminaire organisé conjointement par l'AFUL et la MTIC[98].

Action 2 : *développer des logiciels libres*

L'histoire d'Internet montre que développer des logiciels libres permet

95 La DRIRE (Direction Régionale de l'Industrie, de la Recherche et de l'Environnement) est un service déconcentré de l'Etat dont l'une des missions est le développement en région des PMI. Les DRIRE jouent, en collaboration avec d'autres acteurs régionaux (CCI, conseils régionaux, etc.), un rôle important dans la diffusion du commerce électronique auprès des PMI.

96 http://www.locabureau.com/

97 « Linux : premières pistes pour un usage raisonné ». 15 janvier 1999 à la MTIC. Contact : Godefroy BEAUVALLET (gb@mtic.pm.gouv.fr). La MTIC est chargée est chargée du soutien technique pour le développement des technologies de l'information et de la communication dans l'administration.

98 http://www.mtic.pm.gouv.fr/

de se prémunir contre les stratégies de fermeture des standards du marché en imposant très rapidement une norme de communication publique. Nous avons par ailleurs montré au chapitre 7 que développer un logiciel libre peut être rentable dans de nombreuses situations.

Ces deux opportunités peuvent souvent être conjuguées. Quels acteurs pourraient participer à cet effort de développement ?

Diffuser des normes-métier

Les industriels se regroupent au sein d'associations professionnelles ou de centres techniques (bâtiment, textile, etc.). Certains de ces centres développent ou financent le développement de logiciels propriétaires qui sont ensuite vendus aux membres de la profession. Mais l'effort de commercialisation ou de maintenance n'est pas toujours au même niveau que celui déployé par une éditeur de logiciels propriétaires. Il vaudrait mieux, dans l'intérêt même des membres, que ces logiciels soient diffusés sous forme de logiciels libres.

Ce serait en effet la meilleure façon de diffuser des normes communes de travail et de prémunir l'ensemble d'une profession contre les stratégies de *Winner Take All*. La diffusion mondiale d'un logiciel libre permet, avec un budget marketing réduit, d'atteindre rapidement une masse critique d'utilisateurs. L'économie du logiciel libre permet ensuite de garantir le suivi et une plus grande fiabilité voire une commercialisation plus efficace (ex. distributions Linux) que dans le cas d'un logiciel propriétaire diffusé à très petite échelle.

S'affranchir des outils propriétaires

Le même raisonnement peut être tenu par des consortiums souhaitant éviter qu'un effort de développement commun soit menacé par la politique commerciale agressive d'un éditeur. Là encore, les logiciels libres, permettent de pérenniser un investissement au profit de toute une pro-

fession. Le projet de système embarqué ITRON[99] a permis, depuis plus de dix ans, aux industriels nippons de développer des spécifications communes dans le domaine des systèmes embarqués. La plupart des automobiles, des téléviseurs, des machines outils produits tant par les multinationales que les PME nippones utilisent la norme ITRON pour leur système d'exploitation.

Figure 8. Ken Sakamura, leader du projet TRON

En novembre 1998, plusieurs sociétés nippones et américaines décident de financer le développement d'un système d'exploitation libre, eCos, compatible notamment avec la norme ITRON (voir chapitre 3). Plusieurs arguments sont avancés : mutualisation des coûts de développement, meilleure qualité du logiciel libre, contrôle sur le code source, création d'une offre commerciale permettant à des petites sociétés d'innover. Une des causes probables est l'arrivée de Microsoft sur le marché des systèmes embarqués. Cette arrivée menace à la fois les utilisateurs d'ITRON au Japon et les constructeurs de processeurs dans le monde

99 http://tron.um.u-tokyo.ac.jp/TRON/ITRON/home-e.html

Interview avec Ken Sakamura

- Quels sont pour vous les grandes applications des logiciels libres ?

Je pense qu'un système d'exploitation devrait toujours être un bien public permettant à n'importe quelle société, grande ou petite, de développer une application sur le processeur de son choix. Le modèle de logiciel libre d'eCos est très bon car il permet désormais à n'importe qui de développer des applications à la norme ITRON

- Qui a eu l'idée de développer eCos ?

eCos a été développé car il était moins cher pour plusieurs industriels de financer ensemble le développement d'un logiciel libre plutôt que d'acquérir un logiciel commercial et de payer des royalties ou de développer séparément plusieurs logiciels en interne. Le PDG de Cygnus a trouvé que la norme ITRON était très intéressante. Il a donc souhaité participer au développement du premier système d'exploitation compatible ITRON développé aux Etats-Unis.

- Quels sont les principaux concurrents de la norme ITRON ?

ITRON a pour concurrent principal Windows CE.

- Quel est le rôle des universités dans le domaine du logiciel ?

Je pense qu'une université publique est faite pour produire des biens publics tels que des logiciels libres ou des manuels d'enseignement.

Conférence organisée par l'AFUL le 18/11/1998 http://www.aful.org/presse/ecos.html

entier. Le développement d'eCos et la publicité qui lui est faite aux Etats-Unis doit donc aussi être comprise comme une action défensive visant, en faisant d'ITRON un standard mondial, à préserver l'acquis de compétence nippon dans le domaine de l'embarqué.

Valoriser la recherche

La valorisation de la recherche est devenue une priorité gouvernementale. Il semble en effet anormal que les fruits de la recherche publique financée par le contribuable conduisent à des publications mais plus rarement à des produits, des entreprises ou des créations d'emplois en France. D'où l'idée, admise par tous, de favoriser la création de *start up*. Mais, certaines innovations informatiques ne sont pas facilement valori-

sables dans une *start up*. Par exemple, l'amélioration du fonctionnement interne d'un système d'exploitation ne peut être valorisée que de deux façons : en vendant la technologie à un éditeur ou en intégrant cette technologie dans un système d'exploitation libre comme Linux. La première approche présente un inconvénient : le prix de cession à un éditeur en position dominante ne peut qu'être inférieur à la valeur ajoutée liée à l'utilisation de cette technologie. Ceci reviendrait à demander à la collectivité de payer pour l'utilisation d'une technologie qu'elle a elle-même financée. Aussi, il nous semblerait juste de privilégier la diffusion de ce type de technologie de base sous forme de logiciel libre, intégré par exemple à l'offre Linux, et de vendre sous licence non-exclusive l'usage de cette technologie dans un logiciel propriétaire. Ceci suppose la mise en place d'un système de brevet défensif particulier sur lequel nous reviendrons plus loin.

Grands comptes, encore un effort

Les grands comptes, comme les administrations ou les multinationales, pourraient devenir des acteurs majeurs du développement de logiciels libres en internalisant les externalités induites par leurs choix informatiques. Si, au lieu d'évaluer le coût des logiciels en comparant des prix de licence proposés parfois par les éditeurs au centième du prix public, elles prenaient en compte le prix réel des logiciels, c'est-à-dire le prix que doivent payer la majorité de leurs clients ou de leurs fournisseurs, la comparaison entre l'achat de licences et le développement tournerait très souvent en faveur du développement interne.

En diffusant alors sous forme de logiciel libre un produit développé spécifiquement, l'ensemble de la chaîne client-fournisseur bénéficierait globalement de coûts d'équipement en logiciel bien inférieurs. Les frais de maintenance et de développement du logiciel seraient en pratique assurée au *pro rata* de l'usage. Cette situation nous semblerait à la fois plus saine et plus économique que la situation actuelle où ce sont les petits

utilisateurs qui payent, cher, pour les choix des gros.

Garantir l'ouverture des normes

Plus généralement, le succès d'une politique industrielle dans le domaine des NTIC dépend en grande partie du succès des normes de communication développées. Or, le meilleur outil de diffusion d'une norme est le logiciel libre.

Imaginons par exemple que l'on souhaite voir se développer très rapidement des outils et des normes facilitant l'accès des PME au commerce électronique. Nous avons vu plus haut que les dérives du marché des technologies de l'information conduiraient à une situation de mauvaise communication entre produits concurrents et de prix élevés qui peuvent freiner l'accès de certains acteurs aux produits. Une décision intelligente pour accélérer le passage des entreprises au commerce électronique consisterait à faire développer sur financement public une offre de logiciels libres de commerce électronique mettant en œuvre des normes de communication publiques. Cette tâche pourrait être supervisée par un institut de normalisation (ex. AFNOR, W3C, etc.) dans la mesure où les logiciels libres sont autant des outils de diffusion des normes que des outils de définition des normes. Face à cette offre, il ne fait aucun doute, comme nous l'avons vu dans le cas d'Apache, qu'existera une offre commerciale complémentaire. Charge aux instituts de normalisation de vérifier la conformité des offres concurrentes aux normes publiques et aux services de l'administration de recommander uniquement l'usage de produits conformes à ces normes.

Action 3 : favoriser les actions collectives

Le financement d'actions collectives par cotisation ou par l'impôt est déjà pratiqué dans des secteurs industriels traditionnels. De très petits entreprises italiennes dans le secteur du textile, coordonnées par la mairie ou par les réseaux traditionnels de l'Eglise, ont par exemple montré

qu'elles pouvaient, ensemble, proposer des produits et des prix comparables à ceux d'un grand groupe à condition de s'organiser en réseau. On parle alors de district industriel ou de système de production local[100]. Il est tout à fait courant, et accepté, que les pouvoirs publics aident ce type de regroupement dans le cadre de projets gérés de façon associative.

L'écosystème des logiciels libres étant constitué de nombreux acteurs indépendants qui ont parfois du mal à se regrouper, l'action des pouvoirs publics devrait légitimement viser à mettre en réseau les compétences pour renforcer l'écosystème. Les associations, qui jouent un grand rôle dans l'essor des logiciels libres, pourraient par exemple être chargées de répertorier les logiciels libres, les circuits de diffusion, les *success stories*, les compétences disponibles, etc. En aidant ces projets, les pouvoirs publics pourraient les orienter et veiller à ce que les travaux réalisés au niveau local, régional, national ou international puissent être aisément consolidés afin d'offrir au public un accès plus aisé à la richesse des logiciels libres. On peut imaginer de mettre à la disposition des associations des serveurs, de développer un système commun de classement des logiciels et des sources d'information autour des NTIC, de financer la constitution de bases de données, de journées d'initiation, etc.

La défense des logiciels libres mériterait aussi que l'on soutienne la création de fonds de brevets libres défensifs dont nous décrivons ici le principe. Aux Etats-Unis et au Japon, un logiciel libre est aujourd'hui à la merci de tout dépôt de brevet offensif par un éditeur de logiciel commercial qui contraindrait l'auteur à retirer son logiciel ou l'utilisateur à payer une redevance au propriétaire du brevet pour l'exploiter. Une plainte pour violation de brevet peut ruiner un auteur américain ou japonais et empêcher un auteur européen de faire de son logiciel un standard mondial. Un droit d'usage élevé d'un brevet peut aussi détruire l'avantage économique du logiciel libre. Il s'agit d'une limitation considérable du pouvoir des logiciels libres comme outil de régulation de la concurrence ou

100 http://www.fr-it.com/francital/districtitfr.html

comme base commune pour l'innovation.

Face à une plainte pour violation de brevet, la réaction la plus efficace est de contre-attaquer en déposant en retour une plainte pour violation d'un brevet que l'on possède. Encore faut-il disposer d'une collection de brevets défensifs suffisamment importante pour pouvoir déposer plainte dans n'importe quel cas. D'où l'idée d'inciter les instituts de recherche publics, les universités ou les utilisateurs de logiciels libres à déposer et à mettre en commun les brevets qu'ils céderaient à un fond international de défense des logiciels libres. Pour éviter que ces brevets défensifs ne puissent être utilisés contre les logiciels libres, une clause contractuelle spécifique permettrait à n'importe quel auteur de logiciel libre d'obtenir automatiquement et gracieusement une licence d'exploitation d'un brevet. Les éditeurs de logiciels propriétaires les plus hostiles aux logiciels libres devraient en revanche s'acquitter d'un droit d'usage servant par exemple à financer les frais de dépôts et les frais d'avocat engagés par le fond pour répondre aux poursuites engagées par les entreprises contre les auteurs, ou les éditeurs, de logiciels libres.

EN BREF – A condition de jouer le rôle de standard du marché ou de référence, les logiciels libres permettent d'éviter la domination d'un éditeur au détriment des autres et le choix d'options tarifaires abusives. La concurrence provenant d'éditeurs de logiciels innovants reste cependant possible dans un cadre de respect des normes et d'interopérabilité.

Mais pour jouer leur rôle de régulateur, les logiciels libres doivent être utilisés. Administrations et grands comptes doivent encore faire un effort pour développer l'usage des logiciels libres en leur sein et favoriser leur diffusion. Quant aux pouvoirs publics, leur rôle est certainement de favoriser les actions collectives dans l'écosystème du logiciels libre afin de rendre cette richesse accessible au plus grand nombre et de la protéger contre les tentatives d'appropriation.

9

Une opportunité pour l'Europe

Les logiciels libres agissent comme un régulateur naturel du marché dans une industrie secouée de façon récurrente par des crises monopolistiques. En garantissant l'existence de normes de communication publiques largement adoptées, ils favorisent l'interopérabilité entre produits concurrents, stimulent la diversité de l'offre, modèrent les prix et définissent des standards de qualité à dépasser. Et la possibilité de définir en Europe le contenu des distributions Linux en combinant logiciels libres et logiciels propriétaires offre les moyens de s'affranchir de la domination commerciale américaine dans l'industrie du logiciel.

Linux et les logiciels libres sont donc une véritable opportunité pour les européens. Ils permettent aux consommateurs de s'affranchir des standards propriétaires que certains grands éditeurs américains tentent d'imposer. Ils conduisent à des créations d'emplois plus nombreuses dans les services aux entreprises. Ils garantissent aux éditeurs européens

de logiciel, qui sont souvent des PME, de pouvoir luter à armes égales avec les géants du secteur. Et le droit européen, en excluant la brevetabilité des logiciels et en garantissant une protection élevée des consommateurs, définit un cadre juridique bien plus favorable au développement des logiciels libres qu'au Japon ou aux Etats-Unis.

Mais, sous la pression américaine, l'Union Européenne envisage d'étendre aux logiciels le champ de la brevetabilité. Une telle mesure, en limitant considérablement le pouvoir de régulation économique des logiciels libres, irait dans un sens contraire à l'intérêt général des européens puisque la plupart des grands éditeurs de logiciels sont américains alors que les éditeurs de logiciels européens sont des PME trop petites pour bénéficier en pratique d'une quelconque protection par le brevet. Autoriser la brevetabilité des logiciels pourrait accélérer les transferts d'emplois vers les Etats-Unis et renforcer la dépendance européenne dans les nouvelles technologies de l'information et de la communication.

Une action publique vigoureuse en faveur de la concurrence et de l'indépendance européenne dans l'industrie du logiciel nous semblerait plus conforme aux intérêts des européens. En effet, dans un marché peu concurrentiel, les politiques nationales de développement de la demande conduisent à subventionner les producteurs… américains ce qui limite leur effet créateur d'emploi en Europe. Quant aux politiques d'aide à la création d'entreprises, elles conduisent à subventionner le développement de technologies qui, faute de repreneur européen, sont souvent cédées à de grands éditeurs américains sans garantie sur la pérennité des emplois initialement créés.

S'affranchir de la domination commerciale américaine et du poids de ses oligopoles est une nécessité pour que le développement de la société de l'information soit conduit de façon harmonieuse dans l'intérêt de tous. Aussi, nous proposons deux principes pour guider l'action publique européenne dans le domaine du logiciel : « garantir l'interopérabilité » et « protéger les logiciels libres ».

Une opportunité pour les consommateurs

A niveau de service équivalent, plus de 70% d'économies pourraient être réalisées immédiatement par les entreprises et les administrations sur leur poste « investissement bureautique » à condition qu'elles s'équipent de solutions logicielles et matérielles à base de Linux. 90% d'économies pourraient être réalisées si l'on menait une politique ambitieuse de développement et de diffusion gratuite de logiciels libres et de normes publiques pour la bureautique. Ces économies représentent typiquement, pour un grand grand groupe, de l'ordre de 0,6% de son chiffre d'affaires, pour une PME, de l'ordre de 1,2% et pour une administration locale, de plus de 2,5%[101].

Ce que signifient en pratique ces chiffres est, qu'à budget égal, de nombreuses PME et collectivités locales, qui payent leurs logiciels propriétaires au prix fort et n'ont pas les moyens de proposer une formation digne de ce nom à leurs utilisateurs, pourraient, à condition de faire jouer la concurrence, proposer une formation de qualité à leurs utilisateurs et s'offrir les services d'un ingénieur pour développer des applications spécifiques. Le jeu de la concurrence, et notamment l'usage des logiciels libres, conduirait à des gains notables de compétitivité des PME européennes et à une amélioration du service offert par les collectivités locales grâce à un meilleur usage de la micro-informatique. Par ailleurs, les logiciels propriétaires et les matériels sont souvent importés alors que la formation et les travaux de développement font appels à des compétences locales. Le jeu de la concurrence, et notamment l'usage des logiciels libres, induisent des transferts d'emplois des pays producteurs de logiciels propriétaires et de matériels vers les sociétés de service des pays utilisateurs.

---

101 « Les coûts de la bureautique d'entreprise », Jean-Paul Smets-Solanes, http://smets.com/it/tco/cout_total_invest.html

Une opportunité pour les éditeurs

La fermeture des formats et des protocoles des logiciels propriétaires fait beaucoup plus de tort aux éditeurs de logiciels européens que l'existence éventuelle d'un logiciel libre concurrent. Elle conduit en effet les clients à choisir un logiciel en fonction de sa capacité à lire un format de fichier et non en fonction de qualités techniques, économiques et de caractéristiques innovantes alors que les logiciels libres garantissent le respect de normes communes et stimulent la concurrence.

Lorsque la fermeture est partielle ou lorsqu'existent des solutions pour traduire automatiquement les formats de fichiers des logiciels propriétaires, le client continue à faire ce choix par peur d'être incompatible avec les autres produits. Les grands perdants à ce jeu sont non seulement les consommateurs, qui se retrouvent avec un choix au lieu d'une centaine, mais aussi les industriels européens des nouvelles technologies de l'information et de la communication (voir figure).

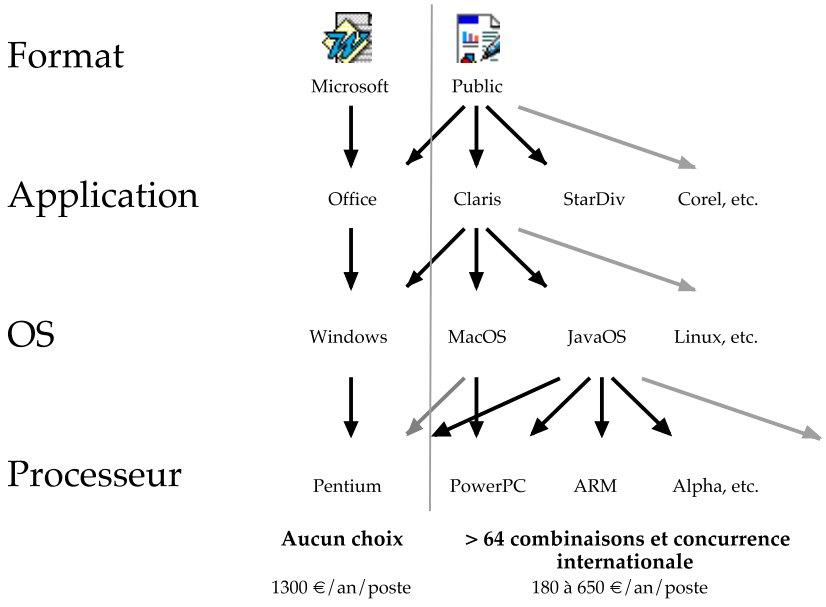

Figure 9. L'ouverture des formats entraîne l'ouverture du choix

Les constructeurs européens de matériel, les producteurs de microprocesseurs et les éditeurs de logiciels sont en effet, majoritairement, des petites entreprises minoritaires sur le marché et disposant de produits le plus souvent supérieurs à la concurrence des grands éditeurs américains. L'anglais ARM conçoit par exemple un microprocesseur à basse consommation particulièrement performant et peu onéreux. Ce processeur a été choisi par de nombreux constructeurs de *Network Computer*, dont l'anglais Acorn, par le constructeur d'ordinateurs de poche Psion et par Corel pour ses serveurs Linux. L'allemand Star Division édite quant-à-lui la seule suite bureautique haut de gamme disponible sur tous les systèmes d'exploitation.

Les exemples de succès techniques et commerciaux de petites sociétés européenne sont très nombreux[102]. Mais Windows 95 ne fonctionne que sur des processeurs compatible Intel et Star Division ne bénéficie pas de toutes les informations techniques sur les produits bureautiques Microsoft ce qui entraîne un délai d'un an avant que Star Office ne devienne compatible avec la dernière version de Microsoft Office.

La concurrence entre PME européennes et grands éditeurs américains est donc faussée par la fermeture des standards du marché lorsque ce sont des standards propriétaires.

En revanche, lorsque des logiciels libres définissent les standards du marché, des sociétés européennes peuvent entrer en compétition plus facilement. Par exemple, l'un des meilleurs éditeurs de sites Web du marché est un logiciel allemand : Golive Cyberstudio[103]. Comme la norme HTML, utilisée par la plupart des sites Web, est une norme publique définie par un logiciel libre du W3C, la concurrence entre logiciels d'édition est possible. Les allemands de Golive, qui bénéficient à Hambourg d'ingénieurs stables et de grande qualité, ont pu facilement s'imposer aux

102 « Des européens qui réussissent ». Jean-Paul Smets-Solanes et Alain Bourissou. Rapport réalisé pour le compte d'INRIA Transferts. Publié sur www.europe-inside.com

103 http://www.golive.com

Etats-Unis dans un environnement très concurrentiel. Les suédois de Roxen sont eux parvenus à détrôner, d'un point de vue technique, le serveur Apache sur le marché des serveurs Web.

EuroLinux facilite l'accès au marché

Les distributions Linux conçues en Europe (SuSE, Mandrake, etc.) peuvent jouer un rôle essentiel pour faciliter l'accès des ces produits au marché. SuSE a par exemple été élu meilleure distribution de l'année par les lecteurs américains du Linux Journal[104]. Or SuSE est livré avec MySQL, Star Office et KDE, des logiciels qui ont aussi été élus meilleurs de leur catégorie par ces mêmes lecteurs. L'effet de synergie entre petits éditeurs européens, illustrés dans cet exemple, peut être exploité pour faciliter l'accès de nouveaux produits européens au marché américain et au marché mondial. Car le phénomène de « *not invented here* » (pas inventé ici) ressenti aux Etats-Unis agit souvent au détriment des petits éditeurs européens qui n'ont pas les moyens d'ouvrir une filiale aux Etats-Unis et de se faire passer pour une société américaine.

La présence en Europe d'acteurs majeurs des distributions Linux facilite l'accès de ces petits éditeurs à un canal mondial de distribution de logiciels et résout en même temps l'une des grandes faiblesses de l'industrie européenne du logiciel : l'absence d'acteur majeur capable de racheter les *start up*. Les éditeurs de distributions européens jouent en partie le même rôle que le consortium Airbus dans leur capacité à rassembler des qualités qui, individuellement, n'ont pas assez de puissance pour s'imposer sur la scène mondiale.

Ce principe de distribution européenne pourrait d'ailleurs être étendu à d'autres systèmes d'exploitation que Linux. Il s'agirait de proposer pour un prix modique (quelques centaines de francs) une collection de logiciels libres (ex. serveur Web, outils de développement, bases de données,

104 1998 Editor's Choice Awards. Marjorie Richardson. Linux Journal 57 - January 1999

Une *success story* européenne: SuSE

- Comment avez-vous a été conduit à fonder SuSE ?

Roland Dyroff, PDG et co-fondateur de SuSE: « Nous étions plusieurs étudiants à l'Université d'Erlangen à vouloir créer une entreprise dans le domaine du service. Nous aimions utiliser Linux. Alors, pour démontrer notre compétence Unix, nous avons conçu une distribution sur CD-ROM en espérant recevoir des contrats de service en retour. Notre CD-ROM a tellement bien marché que nous n'avons finalement jamais eu le temps de travailler dans le service. Nous disposons aujourd'hui de 3000 points de vente en Allemagne et d'une filiale aux Etats-Unis. »

- Quelles ressources consacrez-vous au développement de logiciels libres ?

« Nous employons 4 personnes à plein temps pour développer les pilotes X de Linux. Nous sponsorisons également des projets de logiciels libres comme le projet KDE. »

- Quel est le marché visé par SuSE ?

« SuSE est une distribution Linux généraliste. Nos clients sont à la fois les grands comptes, l'enseignement ou des particuliers. Jusque très récemment, le marché du service pour Linux était peu attractif car les utilisateurs de Linux étaient souvent des spécialistes d'Unix qui n'avaient pas besoin de service. Bien entendu, nous avons des contrats avec des grands comptes. Mais maintenant, Linux est utilisé par des PME qui auront besoin de service. Il s'agit probablement d'un marché très important pour Linux et pour SuSE. »

- SuSE intègre de nombreux logiciels européens…

« De nombreux éditeurs nous proposent d'intégrer à notre distribution des versions de démonstration de leurs logiciels. Cela leur permet d'atteindre une clientèle importante dans le monde. Leurs ventes connaissent alors une forte croissance. Nous sommes d'ailleurs distributeurs officiels de certains de ces produits comme par exemple le SGBD Adabas ou la solution de commerce électronique VShop. »

administration de réseau, etc.) et de les compléter par un ensemble de logiciels propriétaires fournis en version limitée (ex. bureautique, illustration, etc.). Les éditeurs de distributions Linux trouveraient là un moyen à la fois d'étendre leur marché à d'autres plates-formes et d'attirer les utilisateurs d'autres plates-formes vers Linux. Les petits éditeurs européens, qui sont souvent capables de fournir une offre multi-plate-forme, trouve-

raient dans ces distributions multi-plate-forme une entrée rapide et universelle au marché mondial des logiciels. Le consommateurs bénéficieraient d'une offre de type *one stop shop* adaptée à tous les environnements et donc susceptible de préserver au mieux leur investissement.

Garantir l'interopérabilité

Lorsque des logiciels libres servent à définir les standards du marché, la concurrence fonctionne à nouveau de façon raisonnable autour de normes communes ouvertes à tous. Mais les standards du marché sont parfois définis par des logiciels propriétaires et leurs spécifications tenues secrètes par leur éditeur. C'est le cas notamment des outils bureautiques de Microsoft dont les concurrents, faute d'informations suffisantes, ont du mal à exploiter les fichiers alors qu'il s'agit d'une fonctionnalité élémentaire d'un point de vue technique.

S'il est possible en théorie, par décompilation, de découvrir ces standards (formats de fichiers, normes de communication, interfaces de programmation), ce procédé reste suffisamment long et coûteux pour interdire le développement d'une offre de produits concurrents compatibles. L'inertie du marché contribue alors à rendre moins efficace la régulation de la concurrence par des logiciels libres fondés sur des normes ouvertes mais incapables d'exploiter les normes standard du marché.

Aussi, la stratégie de certains éditeurs consiste à détourner progressivement les normes publiques par la tactique « *embrace and extend* »[105] (adopter et étendre) afin d'imposer au marché des normes qui n'ont plus rien d'ouvert ou de public. Par exemple, la plupart des logiciels Microsoft produisent un code HTML qui n'est pas conforme aux recommandations du W3C ou utilise les champs de commentaire pour définir un méta-langage inexploitable par d'autres logiciels. Ce type de pratiques n'est

105 Op. Cit. Halloween

que très rarement sanctionné par le marché lorsqu'il émane d'un acteur dominant.

Face à ce type de stratégies anti-concurrentielles, une action de la puissance publique s'impose pour préserver la diversité sur le marché et renforcer le rôle de régulation de la concurrence des logiciels libres. Deux mesures nous semblent envisageables : utiliser d'une part l'effet d'entraînement des marchés publics pour contraindre les éditeurs à publier les standards utilisés par leurs logiciels et faire appel d'autre part à la certification pour protéger les consommateurs contre les mises en œuvre déviantes de normes de communication publiques.

Proposition 1 : adapter le code des marchés publics

L'introduction des logiciels et des réseaux dans le fonctionnement quotidien de l'administration conduit à remplacer progressivement le support papier par des fichiers numériques. Les logiciels et les réseaux numériques permettent des gains de productivité et de rapidité du traitement administratif. Mais, contrairement au papier, les fichiers numériques n'offrent pas aujourd'hui la garantie de pérennité du papier et d'universalité d'une langue. Il arrive même que l'usage de fichiers numériques engendre des difficultés de communication entres services administratifs et force l'usager à acquérir des logiciels onéreux.

Par exemple, lors de la fusion entre le ministère français de l'industrie et celui de l'économie et des finances, l'une des premières décisions du cabinet fut de remplacer la suite bureautique Corel (utilisé à l'industrie) par celle de Microsoft (utilisée aux finances) car les deux produits avaient du mal à communiquer : les rapports écrits avec un logiciel étaient difficile à lire avec l'autre. L'Union Européenne, qui utilise aussi la suite bureautique de Microsoft, impose *de facto* aux chercheurs souhaitant bénéficier de ses programmes d'aide d'envoyer des fichiers aux normes propriétaires de Microsoft ce qui les contraint à acquérir la suite Office. Plus

préoccupant, de nombreux services administratifs (DDE, DRIRE, etc) sont en train de saisir des données dans des systèmes d'information géographiques régionaux (ex. MapInfo) sensés faciliter l'accès du citoyen à l'information administrative et favoriser la collaboration entre services. Mais personne ne sait si les données saisies aujourd'hui pourront être récupérées dans cinq ans, dans dix ans ou dans cinquante ans car les logiciels employés font appel à des normes propriétaires dont rien ne garantit la pérennité. Toutes ces données sont en effet à la merci d'un changement de version du logiciel ou d'une disparition de l'éditeur et comme leur format est inconnu, la transition vers un autre logiciel pourrait s'avérer difficile.

Toutes ces situations montrent que l'Etat ne joue pas toujours en faveur de la concurrence et de l'ouverture dans l'industrie du logiciel. Mais ces situations sont surtout contraires à deux grands principes : la pérennité des données administratives et le libre accès du citoyen à l'information publique. Aussi, nous proposons de traduire dans le code des marchés publics[106] ces deux grands principes.

Pour éviter les écueils des normes propriétaires et faciliter la maintenance des logiciels dont les fournisseurs auraient disparus, nous proposons d'étendre à tous les logiciels achetés par l'administration le principe de divulgation du code source déjà retenu pour les progiciels[107]. Les modalités de cette divulgation doivent être précisées. Le code source pourrait être remis sous clause de confidentialité à une agence de certification des logiciels (ACL) qui serait aussi chargée de vérifier que les logiciels utilisés par l'administration ne contiennent pas de pièges. Par extension du droit européen sur la décompilation, cette agence aurait aussi le droit d'utiliser le code source des logiciels déposés à des fins d'interopérabilité.

Par ailleurs, pour assurer le libre accès de l'usager aux données admi-

106 http://www.smets.com/it/loi

107 Les Marchés publics d'informatique, par Richard Ganem et Pascal Bouret, collection « Que sais-je ? », Presses universitaires de France.

nistratives, il convient que les normes de communication utilisées par l'administration ne fassent pas l'objet de brevet[108] et que l'accès à ces normes ne fasse pas l'objet de droit d'auteur. C'est ce que nous appelons une norme de communication publique. Une liste de normes publiques officielles serait définie par une communication pour les applications les plus courantes : texte, tableur, image, son, données formatées etc. Seuls les logiciels dont les normes de communication peuvent être traduites de façon exhaustive et au moyen d'un logiciel libre en normes publiques officielles seraient autorisés dans les échanges de l'administration avec ses usagers.

L'interdiction des normes propriétaires pourrait être étendue aux logiciels les plus couramment utilisés dans l'administration, notamment ceux faisant l'objet de commandes nationales, afin de faciliter le jeu de la concurrence entre fournisseurs.

L'ACL serait alors chargée de vérifier la conformité des logiciels aux normes publiques officielles avant tout achat. L'ACL pourrait en outre diffuser des logiciels libres permettant d'accéder librement à l'information numérique publiée par l'administration. Car si l'on considère que certains logiciels élémentaires sont un bien public au même titre que la radio et la télévision publique, il serait tout à fait souhaitable que l'Etat participe au développement d'un corpus minimum de logiciels libres permettant d'assurer aux citoyens une égalité face à l'information administrative numérique.

Proposition 2 : renforcer la protection du consommateur

Les activités de l'ACL pourraient être étendues afin d'assurer une meilleure protection du consommateur face aux dérives de certains éditeurs qui ne respectent pas les normes de communication publiques.

L'agence de certification des logiciels serait chargée d'évaluer la con-

108 C'est automatiquement le cas en Europe car les brevets sur les logiciels sont interdits (voir chapitre 5) mais une directive européenne en préparation menace cet acquis.

formité des logiciels aux principales normes publiques (ISO, RFC, IEEE, W3C, etc.). Son rôle serait triple : développer des tests de conformité sous forme de logiciels libres, publier les résultats de ces tests sur les principaux logiciels du marché et lancer des poursuites contre les éditeurs qui affirment respecter une norme tout en la détournant. L'objectif de l'ACL serait de systématiser à toute l'industrie du logiciel le processus juridique qui a permis aux Etats-Unis à Sun de contraindre Microsoft à se conformer à la norme ISO Java dans l'intérêt des consommateurs.

Les activités de l'ACL faciliteraient le travail d'évaluation des logiciels menés par les entreprises. Par exemple, la plupart des grands comptes commencent par comparer les divers logiciels de messagerie avant de faire un choix *corporate*. Parmi les critères d'évaluation, on peut citer : gestion des accusés de réception, listes de diffusion, sécurité, interface avec le courrier sur Internet, etc. Mais les paramètres à tester étant nombreux, les combinaisons de paramètres ne sont que rarement testées (ex. les accusés de réception seront-ils correctement gérés à travers l'Internet ?). Et ce n'est qu'après quelques mois d'utilisation que l'on découvre les nombreux défauts de logiciels que l'on croyait avoir correctement évalués.

Dans l'exemple de la messagerie, la batterie de tests de l'ACL aurait permis de systématiser la procédure de test et de ne l'appliquer qu'aux rares logiciels dont le résultat des tests n'est pas déjà publié. Par ailleurs, le pouvoir régalien de l'ACL offre au consommateur les moyens de contraindre un fournisseur à mieux respecter les normes qu'il affirme mettre en œuvre. Ainsi, une agence comme l'ACL aurait aussi permis de contraindre Lotus à mieux respecter les normes de l'Internet dans sa messagerie CCMail et d'éviter à ses utilisateurs, notamment dans l'administration française, de recevoir depuis des années des messages d'Internet dans lesquels une partie des adresses électroniques a disparu ou a été remplacée.

Proposition 3 : adapter le droit d'auteur

Au delà d'un meilleur arbitrage entre consommateurs et producteurs, les logiciels libres posent la question de la libre copie des logiciels : vaut-il mieux, l'autoriser ou l'interdire dans l'intérêt général ?

Le droit de copier le code source d'un logiciel, de s'inspirer d'un procédé ou d'une idée offre aux utilisateurs les moyens d'améliorer eux-même et en permanence les logiciels et de diffuser immédiatement ces améliorations. Les entrepreneurs-utilisateurs peuvent assembler ces ressources libres pour proposer au marché de nouvelles solutions. Les chercheurs disposent d'une base de travail ouverte pour implanter et valoriser leurs travaux. La liberté de copier est donc parée de très nombreuses vertus dont nous ne pensons pas que l'on puisse se passer. Une grande partie du patrimoine culturel mondial, des récits médiévaux aux arts graphiques de l'Asie, a d'ailleurs été bâtie grâce à la liberté de copier ou de s'inspirer.

Nous pensons également que la protection des logiciels par le droit d'auteur et l'effort de pédagogie mené auprès des utilisateurs pour les inciter à acquérir des copies légales ont permis l'émergence de l'industrie moderne du logiciel grand public. Des produits tels que les jeux vidéo, dont le coût de production approche celui d'un long métrage, ne peuvent exister que s'ils sont peu copiés et permettent à leur éditeur de rentabiliser rapidement le risque financier qu'ils ont pris. Dans le domaine de l'informatique musicale, le recours systématique à des systèmes physiques de protection contre la copie a par exemple permis à l'éditeur allemand Emagic[109] de garantir le financement et le développement d'un logiciel de composition très innovant alors qu'il était usuel dans la communauté des musiciens de ne pas acquérir les logiciels et de s'échanger des copies.

En ce sens, la protection des logiciels par le droit d'auteur crée un environnement économique favorable aux entrepreneurs. Les utilisateurs

109 http://www.emagic.de

en bénéficient indirectement à travers les produits innovants ou originaux qui émergent parfois.

Lorsque les normes de communications sont ouvertes, l'arbitrage entre ces deux droits peut être régulé par le marché : un manque d'innovation, des prix trop élevés ou un défaut de qualité des logiciels propriétaires favorise l'émergence d'une concurrence à base de logiciels libres plus économiques. L'éditeur d'un logiciel commercial ne peut alors subsister qu'à condition de proposer une offre saine au marché. Or, dans de nombreux domaines, une offre saine s'apparente à une offre de logiciels libres (commerciaux ou non). Publier le code source, permettre aux utilisateurs de contribuer à un logiciel voire autoriser des copies sans payer de *royalties* sont d'ailleurs devenues des caractéristiques de plus en plus fréquentes dans les logiciels commerciaux. Les exemples de la base de données MySQL ou de la librairie graphique Qt montrent que de plus en plus d'éditeurs cherchent à proposer à leurs clients des contrats de licence combinant les avantages du logiciel commercial et du logiciel libre.

Propriété intellectuelle chronodégradable

Aussi, en accord avec notre éditeur, nous avons nous-même appliqué à cet ouvrage un principe de droit d'auteur intermédiaire que nous qualifions de « propriété intellectuelle chronodégradable ». D'une part, notre éditeur souhaite bénéficier d'une exclusivité pour couvrir le risque d'édition pendant la durée de vie probable de cet ouvrage, soit environ un an. D'autre part, certaines parties de cet ouvrage, notamment le guide pratique et les annexes, doivent pouvoir être mises à jour régulièrement pour rester utiles. Nous avons donc décidé, pour faciliter sa diffusion et sa mise à jour à moyen terme, d'abandonner progressivement nos droits d'auteurs, ce qui n'empêche pas notre éditeur de protéger son investissement. Nous avons par ailleurs décidé, à la fois dans l'intérêt général et pour assurer la promotion de l'ouvrage, de renoncer immédiatement à notre droit d'auteur sur certaines parties.

Il nous semblerait juste d'appliquer ce principe de « propriété intellectuelle chronodégradable » à de nombreux domaines de l'édition, et notamment aux logiciels qui remplissent des fonctions d'outils. Une protection de cinq ans par le droit d'auteur sur un logiciel de traitement de texte nous semblerait par exemple plus conforme aux temps caractéristique de renouvellement des produits que les soixante-quinze ans qui sont actuellement applicables. En déposant le code source de leurs logiciels à l'ACL et en autorisant les adaptations libres au bout de cinq ans, l'industrie du logiciel bénéficierait d'un cadre juridique plus favorable que celui qui oblige aujourd'hui les constructeurs automobiles à fournir des pièces détachées pendant plus de vingt ans. Ce cadre garantirait aussi aux consommateurs de pouvoir maintenir et de récupérer les données produites par un logiciel devenu non rentable pour son éditeur.

Dans le cas de logiciels à contenu artistique ou intellectuel (CD-ROM, encyclopédie, jeu vidéo, musique etc.), la protection du contenu pourrait être séparée de la protection du contenant afin qu'une encyclopédie conçue pour un système d'exploitation donné puisse être adaptée ultérieurement à un autre système d'exploitation par ses lecteurs. L'éditeur serait seulement contraint de fournir le code source de l'outil de consultation et d'en autoriser les adaptations.

Le principe de « propriété intellectuelle chronodégradable » pourrait aussi être appliqué à des domaines artistiques, comme la musique électronique de type *house* ou *techno*, où la copie et l'emprunt font partie intégrante du processus de création et d'interprétation.

Protéger les logiciels libres

Le plus grand danger pour l'industrie européenne du logiciel est l'appropriation des normes de communication par certains grands éditeurs américains. Les logiciels libres permettent d'éviter cette appropriation en

imposant des normes de communication publiques ou en offrant aux utilisateurs et aux éditeurs les moyens d'exploiter librement des normes de communication propriétaires. Mais les logiciels libres sont menacés aux Etats-Unis et au Japon par le droit des brevets qui pourrait interdire à un auteur de donner un logiciel sans l'autorisation d'un éventuel détenteur de brevet. L'Europe doit-elle suivre la même voie ou proposer un autre modèle pour l'industrie du logiciel ?

Rejeter des brevets nuisibles

Imaginons une situation où un utilisateur/auteur développe un logiciel de compression d'images et le met à la disposition de tous sur Internet. Supposons que, sans le savoir, il viole un brevet ou que, par malchance, un grand éditeur dépose ensuite un brevet en faisant valoir une antériorité sur la publication du logiciel.

Ce scénario est non seulement plausible mais est déjà survenu[110]. L'utilisateur/auteur du logiciel se voit menacé par l'éditeur et, en cas de procès, risque de perdre toute sa fortune personnelle en frais d'avocat. Une telle situation, si elle était considérée comme légale et se généralisait en Europe, conduirait virtuellement à interdire la publication des logiciels logiciels libres et leur utilisation. A moins de disposer d'une importante collection de brevets défensifs, les auteurs de logiciels libres auraient du mal à se protéger légalement contre les agressions des éditeurs de logiciels commerciaux.

Accepter en Europe la brevetabilité du logiciel, comme l'envisage aujourd'hui l'Union Européenne[111], reviendrait à offrir aux grands éditeurs américains les moyens de contrôler l'industrie européenne du logiciel en d'interdisant la diffusion de logiciels libres susceptibles de menacer des profits fondés sur une stratégie de domination du marché. Une telle me-

110 http://www.slashdot.org/articles/98/11/11/0044248.shtml

111 Promouvoir l'innovation par le brevet - Livre vert sur le brevet communautaire et le système des brevets en Europe.

sure nous semble inopportune, tant pour les consommateurs européens qui y perdraient du pouvoir que pour les éditeurs européens qui risqueraient de se faire attaquer par des groupes plus puissants qu'eux.

Qu'en est-il sur le fond ? Les brevets ont été inventés pour inciter les inventeurs à publier leur invention en échange d'un monopole sur l'usage de l'invention. Sur le principe, le brevet permet de protéger l'inventeur tout en évitant que de bonne idées restent secrètes. Mais, dans la pratique, déposer un brevet international coûte près de 100 000 F par an et le défendre en cas de violation coûte plus de 250 000 F. Le dépôt de brevet, et plus encore sa défense, sont souvent hors de portée financière pour les petites sociétés. C'est pourquoi le brevet ne protège, en pratique, que les grandes sociétés. L'échange de brevets entre grands industriels est parfois utilisé pour constituer des oligopoles technologiques et exclure les nouveaux entrants sur un marché. De ce point de vue, le brevet sur le logiciel n'a rien de spécifique.

Les brevets sur les logiciels ont longtemps été rejetés car ils correspondent à des procédés dans l'exercice d'activités intellectuelles et non à des procédés à caractère industriel. Les logiciels sont en effet conçus par application de principes mathématiques ou de principes d'organisation. Or ces principes font partie du patrimoine de l'humanité, ne serait-ce que parce que chacun est libre de les reproduire dans son *for* intérieur sans possibilité de contrôle ou de contrainte externe. En reproduisant mentalement, et non avec un ordinateur, un procédé breveté dans le cas d'un logiciel, on devient soi-même un contrefacteur. En permettant de s'approprier ces procédés, la brevetabilité des logiciels tend *in fine* à remettre en cause la liberté de penser.

Il est aussi particulièrement aisé de découvrir un procédé intellectuel élémentaire brevetable au sein d'un logiciel. Par exemple la principe élémentaire consistant à vendre de la musique téléchargeable sur Internet fait l'objet d'un brevet aux Etats-Unis. N'importe qui est certainement capable d'inventer chaque mois, voire chaque jour, un tel procédé ; l'in-

dustrie du logiciel est encore jeune et ce que l'on présente comme des inventions sont avant tout des découvertes élémentaires. Aussi, pour un investissement en R&D très faible, on peut se constituer une grande collection de brevets. Plus que dans toute autre industrie, la brevetabilité des logiciels est d'abord favorable aux grands industriels et récompense plus une assise financière qu'une réelle capacité d'innovation.

Le dépôt de brevets sur les logiciels est d'ailleurs devenu si important aux Etats-Unis que le bureau américain des brevets est devenu incapable de procéder à des recherches d'antériorité fiables. Un brevet a été récemment accordé à Microsoft pour une technologie de lissage de caractères déjà publiée en 1976. Les brevets sur le logiciel ont donc une valeur aléatoire et ne permettent même pas de valoriser correctement la technologie d'un éditeur de logiciel en cas de fusion-acquisition, en dehors de la capacité de nuisance que procure le brevet à son acquéreur.

Plus généralement, les brevets sont désormais utilisés par les grands éditeurs de logiciels qui, ayant compris que l'ouverture du code source était désormais acquises pour les consommateurs, cherchent un nouveau moyen de perpétuer des stratégies de contrôle et de domination du marché. Par exemple, la licence SCSL autorise sous condition l'adaptation du code source de Java mais interdit implicitement le développement d'un logiciel compatible qui violerait des brevets déposés par Sun autour de Java. L'usage de la technologie Java n'est donc absolument pas libre et ses conditions sont fixées par Sun qui souhaite contrôler le marché en excluant *a priori* toute initiative n'entrant pas dans ce cadre. Bien entendu, tout le volet « brevet » de cette licence est illégal en Europe mais il pourrait, aux Etats-Unis, limiter le succès de technologies concurrentes comme Kaffe (un produit anglais) qui ne dispose probablement pas des moyens financier pour lutter contre Sun alors que sa technologie a été élue meilleure de l'année 1998.

On a découvert récemment que Microsoft avait déposé un brevet sur

une technologie de mise en page[112] qu'elle avait fait adopter auparavant par ses partenaires du W3C[113] sans leur indiquer l'existence d'une procédure de dépôt. Ainsi, un élément clef de la norme officielle du Web appartient désormais à un acteur privé qui a tout loisir d'imposer les conditions qu'il souhaite pour son usage. Ce dernier cas illustre bien le principal danger des brevets sur le logiciel : l'appropriation juridique des normes de communication, dont les conséquence sont encore bien plus inquiétantes que l'appropriation technique des normes par le secret.

C'est pourquoi nous estimons que la brevetabilité des logiciels est très souvent contraire à l'intérêt général et devrait être rejetée ou fortement limitée.

Pour une industrie diversifiée et innovante

Mais sans brevets, une partie minoritaire du modèle américain de *start up* technologique ne fonctionne plus. Dans ce modèle, la capacité de prédation d'un marché est l'un des ressorts avancés pour justifier la possibilité de faire financer par des fonds de capital risque certaines innovations technologiques majeures. La *start up* commence par produire de la propriété intellectuelle originale puis se crée un marché par un investissement marketing considérable avant de prélever une rente sur ce marché captif pour couvrir l'investissement technologique, l'investissement marketing et le risque pris initialement. Comme une innovation technologique majeure met environ dix ans à être acceptée par un marché, il est nécessaire de pouvoir verrouiller la propriété intellectuelle des technologies pendant de nombreuses années. D'où le recours présenté comme nécessaire au droit des brevets et au droit d'auteur pour garantir pendant au moins dix ans un niveau de prix suffisamment élevé pour couvrir l'investissement initial en marketing et protéger l'investisseur.

En revanche, le développement rapide de produits innovants par des

112 http://www.patents.ibm.com/patlist?icnt=US&patent_number=5860073
113 Consortium international chargé de définir les normes du Web.

start up qui jouent la notoriété d'une marque est tout à fait possible et même facilité par le fait que ce type de *start up* ne risque pas d'être accusée de violation de brevet par un concurrent déjà établi sur le même segment de marché. Leur financement par capital risque leur permet d'accéder au marché et de jouir rapidement d'une grande notoriété. Dans la mesure où leurs produits ne modifient pas radicalement les habitudes du marché, elles peuvent devenir rapidement rentables en trois à cinq ans et n'ont pas besoin de protéger par des brevets des prix élevés. Elles peuvent compter sur leur avance et leur capacité d'innovation pour rester compétitives face à d'éventuels développements de logiciels libres par des utilisateurs qui jugerait ces prix trop élevés. Certaines de ces *start up* ont même intérêt à diffuser leurs produits sous forme de logiciels libres, soit immédiatement pour se créer un marché, soit après quelques années pour contrôler le marché qu'elles possèdent.

Comme la très grande majorité des *start up* financées par capital risque développe de nouveaux produits, de nouveaux services et non des nouvelles technologies majeures, on peut affirmer que la suppression des brevets sur le logiciel favorise l'innovation, la création d'entreprises et… les fonds de capital risque. Des sociétés comme Red Hat, Sendmail ou VA Research qui sont spécialisés dans les logiciels libres sont financées aux Etats-Unis par des fonds de capital risque. La société Cobalt, qui a dans son capital le fond de capital risque de France Telecom, a même reçu le prix du produit le plus innovant de l'année par le magazine PC Computing[114]. Aucune de ces sociétés n'a besoin de brevet pour se protéger. Le droit d'auteur et le droit des marques leur suffit.

Reste l'innovation technologique majeure. Elle provient majoritairement depuis plus de trente ans de laboratoires de recherche publics où elle n'est que rarement valorisée ou brevetée. Lernout et Hauspie, l'une des *start up* technologiques les plus célèbres en Europe, a par exemple démarré en utilisant les travaux de recherche d'universitaires belges dans

114 http://www.cobaltmicro.com/company/press/press27.html

le domaine de la reconnaissance vocale avant de devenir l'un des leaders mondiaux des industries de la langue. Aussi, sa contribution à l'état de l'art doit, de notre point de vue, être considérée comme une valorisation de résultats de la recherche publique associée à un effort industriel de constitution d'une base de donnée linguistique, protégeable par le droit d'auteur.

Il existe probablement quelques exemple d'innovations technologiques majeures en informatique provenant du secteur privé comme l'invention des fenêtres et des souris. Mais, même dans ce cas, le brevet a joué en défaveur de Xerox dont l'invention a été considérée par la justice américaine comme une idée et non comme un procédé. Le brevet a été annulé au profit d'Apple qui s'était pourtant inspiré des travaux de Xerox dans le domaine. Le rapport entre les brevets et l'innovation est donc à peu près inexistant dans l'industrie du logiciel. Les brevets freinent les produits innovants sans protéger les grands inventeurs. Il serait donc illusoire de croire que développement de *start up* suffit à faire une politique de valorisation de l'innovation technologique et scientifique et que les brevets sont une condition nécessaire.

C'est pourquoi, il nous semblerait plus raisonnable que l'Union Européenne de finance des projets de développement de logiciels libres innovants dans un cadre public ou associatif[115] et sursoie à sa décision d'instituer un brevet sur le logiciel afin de s'assurer d'abord qu'elle est bien conforme à l'intérêt général de tous les européens.

www.freepatents.org

Nous incitons le lecteur à consulter le site www.freepatents.org pour prendre connaissance en détail des enjeux de la brevetabilité des logiciels et soutenir une action visant à éviter que l'Union Européenne n'adopte

115 Les associations peuvent jouer un rôle primordial dans l'innovation comme le montre l'exemple de Téléthon. Voir « Les stratégies des associations humanitaires dans le domaine de la recherche médicale », Vincent Jonquières et Jean-Paul Smets-Solanes. Ecole des Mines de Paris. 1999.

une directive contraire aux intérêt. des européens.

La brevetabilité des logiciels est une question complexe. De nombreux experts européens[116], influencés par des phénomènes de mode ou poussés par des intérêts particuliers, ne semblent pas avoir pris la mesure des spécificités de l'industrie du logiciel. Alors que des sociétés américaines comme Oracle affirment que les brevets sur les logiciels constituent un système juridique inadapté, coûtant très cher et ne rapportant rien, les experts de l'Union Européenne croient favoriser l'industrie du logiciel en lui offrant un système de brevets.

Citant un manque d'innovation en Europe pour justifier leur position, ces mêmes experts mettent en jeu la vie de centaines d'éditeurs européens particulièrement innovants, qu'ils ne connaissent d'ailleurs pas. Enfin, alors que le procès Microsoft ne cesse de révéler certaines pratiques anti-concurrentielles dans l'industrie du logiciel, les experts de l'Union Européenne présentent Microsoft comme l'exemple à suivre en matière de brevets et d'innovation.

EN BREF – Linux et les logiciels libres offrent aux européens les moyens de garantir une situation de concurrence sur le marché des NTIC et de s'affranchir de la domination des grands éditeurs américains. Mais des pressions sur l'Union européenne menacent ces deux opportunités, par l'introduction de brevets sur les logiciels, sans pour autant stimuler l'innovation.

Deux principes son proposés pour guider l'action publique européenne dans le domaine du logiciel : « garantir l'interopérabilité » par l'ouverture des normes et « protéger les logiciels libres ».

116 Voir par exemple http://www.education.gouv.fr/actu/assisinn/DATA/LOMBARD.HTM.

La nouvelle économie

Lors de chaque révolution informatique, il y a des gagnants et des perdants. Microsoft est parvenu à renverser IBM dans les années 80 pour avoir compris suffisamment tôt l'importance de la micro-informatique. Mais c'est aujourd'hui IBM qui semble le mieux comprendre l'impact de la révolution du logiciel libre et menace -avec d'autres- la suprématie de Microsoft.

Le modèle technique et économique du logiciel libre, notamment l'approche *open source*, est une tendance de fond qui vise à donner à l'utilisateur plus de contrôle et plus de responsabilité sur les logiciels qu'il utilise. Il s'agit donc de la suite naturelle de la révolution qui, il y a dix ans, a permis à chaque utilisateur de disposer d'un ordinateur personnel. Comme la micro-informatique, le logiciel libre devrait à la fois conduire à une réduction des coûts, à rendre certaines technologies plus accessibles et à susciter l'émergence de nouveaux services.

Les premiers gagnants de la révolution des logiciels libres sont les utilisateurs qui peuvent faire jouer plus facilement la concurrence entre fournisseurs de matériel, de logiciel et de service. Dans l'industrie informatique, les grands vainqueurs pourraient être les sociétés qui parviennent à libérer le code source de leurs logiciels tout en conservant le con-

trôle de l'évolution de leur produit et des ventes de matériel ou de services autour de leurs produits. Les petits éditeurs de logiciels, qui sont très nombreux en Europe, sortiront aussi gagnants car ils peuvent entrer plus facilement sur le marché grâce à l'existence de normes de communication communes et de technologies de base facilement accessibles sous forme de composants logiciels libres. Leurs produits, souvent très performants et peu onéreux, trouvent facilement une place à côté d'une offre de logiciels libres. La diffusion des logiciels libres engendre ainsi une création accrue d'emplois en Europe, emplois qui étaient auparavant créés outre-Atlantique.

Les grands perdants sont ceux qui, parmi les éditeurs, ne prennent pas assez en compte l'intérêt des utilisateurs et scient la branche sur laquelle ils se trouvent. Ces derniers favorisent le développement d'une offre concurrente sous forme de logiciels libres qui jouent dans ce cas un rôle bénéfique de régulation des pratiques abusives.

L'Europe, qui ne reconnaît pas les brevets sur les logiciels, bénéficie là d'une opportunité pour devenir leader de cette révolution. Encore faudrait-il que ses fonctionnaires de l'Union Européenne fassent la promotion internationale du droit européen du logiciel et du modèle européen de politique de l'innovation au lieu d'étudier les façons de transcrire en Europe un droit américain contraire à l'intérêt général, notamment à celui des européens.

Il en va de l'intérêt de chaque citoyen européen. Car le logiciel libre n'est pas seulement affaire de liberté et d'équité pour les entreprises. Outre-Atlantique, il a parfois été confondu, à tort, avec l'ultra-libéralisme de « l'Idéologie californienne »[117]. Mais en Europe, son acclimatation peut

117 L'Idéologie californienne, très présente dans la Silicon Valley, préconise un désengagement maximal de l'Etat au profit de l'initiative privée et de l'autorégulation des citoyens. Elle reprend les idées du petit Parti Libertarien, mélange d'anarchisme de droite pour le social et d'ultra-libéralisme pour l'économie. Certains membres de la Free Software Foundation (FSF) de Richard Stallman sont proches de ce parti, parce qu'ils contestent notamment la notion de propriété intellectuelle. Pourtant, rien n'est plus contraire à cette idéologie que les statuts de la FSF, qui défend précisement le droit d'auteur. Pour en savoir plus sur l'Idéologie californienne : « Californian Ideology », Richard

être un puissant facteur d'égalité, alors que l'accès aux technologies de l'information devient de plus en plus indispensable à tous. Parce qu'il est gratuit, parce qu'on peut l'adapter à ses besoins sans facture supplémentaire, il permet aux petits budgets de s'informatiser de manière pérenne : écoles, associations, collectivités locales rurales, PME, etc. Ce qui évite de faire grandir le fameux fossé entre info-riches et info-pauvres. Car créer de la richesse ne doit pas signifier créer de la pauvreté.

Toutefois, il ne faut pas confondre *copyleft* (gauche de copie) et logiciel gauchiste. Le logiciel libre ne doit pas nourrir les mêmes illusions que l'Internet lors de son explosion du milieu des années 90, lorsque certains prophètes voyaient dans le réseau mondial un moyen de faire disparaître les conflits sociaux. Le logiciel libre ? Juste une goutte de tempérance dans la rudesse de l'économie de marché.

Barbrook & Andy Cameron, http://ma.hrc.wmin.ac.uk/ma.theory.4.2.1.db

Glossaire

Apache : prononcez « a-patchi ». Serveur Web libre, utilisé par la majorité des sites de la Toile. Conçu par des développeurs -issus notamment du magazine Wired- en 1995, à partir du code source du serveur du National Center for Supercomputing Association (NCSA).

BSD : abréviation de Berkeley System Distribution. Nom du système Unix (voir ce mot) conçu en 1977 à l'Université de Berkeley (Californie). La licence publique BSD étant moins restrictive que la GPL (voir « *copyleft*"), ce premier Unix libre a divergé vers des versions propriétaires, dont celle de BSD Inc., et vers plusieurs variantes libres : NetBSD, FreeBSD et OpenBSD.

Code source : Corps d'un programme composé de milliers (ou de millions) de lignes de programmation. Dans la cas d'un logiciel libre, celui-ci est nécessairement accessible aux utilisateurs (voir » *Open Source* «).

Copyleft : En français, » gauche de copie » ou » copie laissée « . Type de droit d'auteur inventé par Richard Stallman, fondateur de la FSF (voir ce mot). Le *copyleft* permet à tous de profiter de l'usage d'un logiciel, de le modifier et même de le revendre sous forme de logiciel libre (ce qui le différencie du *copyright*) mais interdit à quiconque d'en prendre possession (ce qui le distingue du domaine public).
Le principe du *copyleft* a servi de base à la licence la plus courante pour les logiciels libres, la GPL (voir ce mot).

Distribution : Version packagée de Linux distribuée sous forme de CD-ROM par un revendeur spécialisé. Elle comprend une préconfiguration du système et des outils complémentaires (suite bureautique, serveur Web,...). Les plus connues (SuSE, Red Hat, Open Linux de Caldera) ont fait la fortune des premières *start up* du logiciel libre.

Free Software Foundation (FSF) : créée en 1984 par Richard Stallman, chercheur au laboratoire d'Intelligence Artificielle du MIT, la Free Software Foundation a servi de cadre au développement des outils GNU. Aujourd'hui, elle continue de promouvoir et de diffuser les logiciels libres à travers son charismatique fondateur. D'autres organisations comme Linux International, dirigée par John Hall, senior manager chez Digital-Compaq, ou *Open Source* International, d'Eric Raymond, jouent un rôle comparable.

Freeware : Très souvent confondu avec « logiciel libre ». En anglais, le mot « free » désigne autant la gratuité que la liberté. Bien qu'on utilise parfois ce mot pour les logiciels combinant *Open Source* (voir ce mot) et *copyleft* (voir ce mot), un « *freeware* » peut n'être que « gratuit » -tout en restant sous *copyright* privé. Tandis qu'un logiciel libre peut-être payant. Les « *freeware* » sont complémentaires des *shareware* (voir ce mot).

GNU : nom du projet initié par Richard Stallman, fondateur de la Free Software Software (voir ce mot) en 1983 et destiné à concevoir un Unix libre. D'où le sens de l'abréviation de GNU, « GNU is Not Unix". Le projet GNU a débouché sur la mise au point du principe du « *copyleft* » (voir ce mot), qui sert de base juridique à la majorité des logiciels libres. Surtout, GNU a donné naissance à la plupart des outils libres « historiques » (éditeur de texte Emacs, compilateur gcc, débogueur GDB,…), lesquels ont servi d'environnement de développement pour l'Unix libre Linux (voir ce mot).

GPL : General Public Licence. Licence Générale Publique utilisée pour les outils conçus dans le cadre du projet GNU, mais aussi pour bon nombre de logiciels libres dont Linux. Dans le cas d'un logiciel sous GPL, l'auteur fournit le code de son produit, abandonne ses droits sur celui-ci et autorise donc la libre copie, distribution et modification aux utilisateurs. En contrepartie, tout programme issu d'une modification d'un logiciel sous GPL est lui aussi libre. Il existe d'autres modèles de licences moins exigeant (la licence BSD -voir ce mot)

ou moins » libres » (la Netscape Public Licence, qui permet à l'éditeur de conserver le *copyright* sur le code source).

Hacker : aujourd'hui, le mot « *hacker* » est surtout utilisé pour désigner un pirate qui force un système d'information protégé. Dans la communauté du logiciel libre, il s'agit d'un développeur qui apporte des modifications ingénieuses à un programme. Plus artistes que bandits, Linus Torvalds et Richard Stallman sont un peu les princes des *hackers*.

Library : en français, bibliothèque. Ensemble de routines, sous-programmes et procédures mis à la disposition des programmeurs pour . La majorité des bibliothèques libres sont gérées sous le régime de la General Library Public Licence (GLPL). Contrairement à la GPL (voir ce mot), la LGPL n'exige que les programmes utilisant ses librairies deviennent libres à leur tour.

Linux : Unix libre pour PC dont le développement a débuté en 1991, sous la conduite du Finlandais Linus Torvalds. En plus de la plate-forme Intel, il a été porté depuis sur la plupart des processeurs (Alpha, PowerPC, etc.). Linux a toutes les qualités d'un Unix, le prix en moins : il est gratuit au téléchargement et coûte entre 300 et 500 FF lorsqu'il est distribué en CD-ROM (voir » distribution «).

Open Source : avec source code (voir ce mot) modifiable et accessible à tous. L'un des principes fondateurs du logiciel libre avec le *copyleft*. Ce concept a débouché sur l'*Open Source* Software (OSS), modèle de développement communautaire théorisé par Eric Raymond, auteur de « la Cathédrale et le Bazar » et cofondateur du consortium *Open Source* International (OSI).

OS (operating system) : système d'exploitation, soit la couche logicielle qui permet à un ordinateur de faire tourner ses applications. Windows et Linux sont des OS.

Shareware : En français, logiciels contributifs ou partagiciels. Logiciel téléchargeable ou distribué sur CD-ROM et dont la licence d'utilisation est directement payable à l'auteur, souvent après une période d'essai. Les *shareware* ont pour corrolaire les logiciels gartuits (voir « *freeware* »). Tous deux ne sont pas livrés en *Open Source* (voir ce mot) et ne sont pas régi par le régime du *copyleft* (voir ce mot).

Terminal X : postes clients fonctionnant avec un protocole X11 leur permet-

tant d'afficher des applications exécutées en réalité sur un serveur. Très utilisées dans les solutions client-serveur sous Linux (voir ce mot).

Unix : Uniplexed Information and Computer Service. Système d'exploitation multitâches et multi-utilisateurs conçu en 1969 dans les laboratoires d'AT&T. Unix est réputé pour sa stabilité et sa robustesse, qualités dont ont hérité les Unix libres Linux et BSD (voir ces mots). Les logiciels libres du projet GNU (voir ce mot) étaient d'ailleurs destinés à fonctionner en environnement Unix et devaient à terme permettre de se passer des Unix commerciaux (« Gnu's Not Unix ») pour concevoir un équivalent libre, Hurd. Hurd existe à l'heure actuelle à titre expérimental.

Annexe A
Les essentiels

Sites

www.fsf.org est le site de la Free Software Foundation de Richard Stallman qui développe depuis quinze ans les grands principes juridiques du logiciel libre et a fourni à la communauté les principaux outils de développement qui ont permis à Linux d'exister.

www.opensource.org est le site dirigé par Eric S. Raymond où sont définis les principaux principes de l'approche *Open Source* que l'on peut considérer comme une généralisation du modèle libre adaptée à une plus grande partie de l'industrie.

www.linux.org est le principal site de référence autour de Linux.

www.freebsd.org est le principal site de référence autour de FreeBSD.

pauillac.inria.fr/~lang est le principal site francophone de référence.

Textes fondateurs

La Cathédrale et le bazar par Eric S. Raymond démontre la validité, d'un point de vue d'ingéniérie, du modèle de développement du logiciel libre.

http://www.tuxedo.org/~esr/writings/cathedral-bazaar/

Traduction française :

http://www.eleves.ens.fr:8080/~blondeel/traduc/Cathedral-bazaar/

The Halloween Document est un rapport interne à Microsoft où sont démontés les mécanismes de développement du logiciel libre pour déterminer la meilleur façon des les combattre.

http://www.opensource.org/halloween.html

Articles de référence

« **Vive le logiciel libre** » par Yves Eudes – Le Monde 18/03/96

« **For the love of hacking** » - Forbes 10/08/98
http://www.forbes.com/forbes/98/0810/6203094a.htm

The greatest OS that (n)ever was – Wired, août 1997
http://www.wired.com/wired/5.08/linux.html

Les licences

GNU General Public License (GPL) http://www.fsf.org/copyleft/gpl.html

Traduction non officielle en français de la GPL
http://www.linux-france.com/article/these/gpl.html

GNU Library General Public License (LGPL)
http://www.fsf.org/copyleft/lgpl.html

Aladdin Free Public License
http://www.cs.wisc.edu/~ghost/aladdin/doc/Public.htm

Artistic License http://language.perl.com/misc/Artistic.html

Aladdin License ftp://ftp.cs.wisc.edu/ghost/papers/frs96.txt

Netscape Public License (NPL et MPL) http://www.mozilla.org/NPL/

X Windows System License http://www.x.org/xlicense.html

Open Content http://www.opencontent.org/

Qt Public License (QPL) http://www.troll.no/qpl/

SCSL (Sun Community Source License)
http://java.sun.com/products/jini/licensing/scslprinciples.html

Annexe B
Carnet d'adresses

La presse et l'information

Linux Center (en français) http://www.linux-center.org

Les Nouvelles Neuves de Linux http://www.linux-center.org/news/

Slashdot http://www.slashdot.org/

Freshmeat http://www.freshmeat.com

LinuxHQ http://www.linuxhq.com

LinuxGames http://www.linuxgames.com/

Linux Today http://linuxtoday.com/

Linux Weekly News http://lwn.net/

Linux Resources http://linuxresources.com/

LinuxWorld http://www.linuxworld.com/

Où les télécharger ?

Les supermarchés du shareware

Les sites ci-dessous sont des sites de téléchargement généralistes qui comprennent une proportion de logiciels libres ou de logiciels pour Linux.

filez.com http://www.filez.com

shareware.com http://www.*shareware*.com

download.com http://www.download.com

TUCOWS http://www.tucows.com

Les supermarchés de l'Open Source

Les sites ci-dessous sont spécialisés dans les logiciels libres ou les logiciels pour Linux.

Linux Center (en français) http://www.linux-center.org

Linux Applications http://www.linuxapps.com

Linux File Archive Browser http://www.linuxhq.com/lsm/

Linux Software Map http://linuxresources.com/apps/index.html

NOL Store http://store.nol.fr/

Les éditeurs

Un nombre croissant de sociétés commerciales d'édition de logiciels choisissent de distribuer leurs logiciels sous forme de logiciel libre ou de logiciel semi-libre. Une sélectrion est indiquée ci-dessous.

Cygnus : http://www.cygnus.com

Cygnus édite depuis plus de dix ans des compilateurs libres très largement diffusés dans l'industrie et depuis peu un système d'exploitation libre, eCos.

IBM http://www.ibm.com

IBM diffuse sous forme de logiciel *Open Source* un compilateur Java et un serveur de courrier électronique sécurisé.

Troll http://www.troll.no

Troll édite un bibliothèque de composants logiciels *Open Source* pour la construction d'interfaces graphiques.

SSH http://www.ssh.fi

SSH édite un serveur d'accès sécurisé selon un modèle semi-libre.

TCX http://www.tcx.se

TCX diffuse la base de données MySQL selon un modèle semi-libre.

GoodNoise http://www.goodnoise.com/ et http://www.freeamp.org/

GoodNoise édite un logiciel libre de lecture de fichiers sonores MP3.

Roxen http://www.roxen.com/

Roxen vend une suite complète de serveurs Web sous forme de logiciel libre.

Transvirtual http://www.kaffe.org

Transvirtual édite sous forme de logiciel libre l'une des meilleures machines virtuelles Java.

Netscape http://www.netscape.com et http://www.mozilla.org

Netscape diffuse désormais son navigateur sous forme de logiciel *Open Sourve*.

Apple http://www.apple.com

Apple diffuse sous forme de logiciel libre un environnement Smalltalk très complet.

Zope http://www.zope.org/

Zope édite un environnement de développement d'applications Web et a choisi une diffusion sous forme libre pour accélérer sa croissance.

Sun http://www.sun.com

Sun diffuse une partie des technologies autour de Java sous forme de logiciel libre ou semi-libre.

sendmail http://www.sendmail.com/

Le serveur de courrier électronique le plus utilisé au monde est un logiciel libre

Acheter une distribution

Voir l'Annexe 3 pour une liste de ditributions Linux.

Liste des distributeurs SuSE

http://www.suse.de/reseller/france/france.html

Liste des distributeurs Mandrake

http://www.linux-mandrake.com/contenu/fsinglecd.html

Liste des distributeurs Debian

http://www.debian.org/distrib/vendors

Liste des distributeurs RedHat

http://www.redhat.com/resellers.phtml

Liste des distributeurs Caldera

http://www.calderasystems.com/cgi-bin/relocate?country=FRANCE&title=France

Les solutions

Cobalt Qube http://www.cobaltmicro.com

Serveur Linux préinstallé avec interface graphique conviviale et design agréable. Disponible en France chez Pick Systèmes, Mandala et la plupart des revendeurs Linux. Peut être acheté directement par correspondance.

CXR NetLine 101 http://www.cxt.fr

Routeur, *firewall* et adaptateur numéris sous Linux permettant de partager de façon sécurisée un accès Internet

Les organisations

Linux International http://www.li.org

Les associations en France

AFUL (Association Francophone des utilisateurs de Linux et des logiciels libres) http://www.aful.org

Freenix. http://www.freenix.fr/freenix.html

April (Association pour la Promotion et la Recherche en Informatique Libre) http://www.april.org

Club Virtuel des Utilisateurs de Linux. http://perso.wanadoo.fr/vdefert/

Proselux (volontaires pour une assistance à l'installation)
http://www.linux-france.com/article/proselux/

Associations régionales

Groupe des utilisateurs de Linux du Dauphiné http://www.guilde.asso.fr/

BeLug (BeOs & Linux Users Group - Languedoc)
http://wwwperso.hol.fr/~albruc/index.htm

Provence Linux Users Group http://www.pipo.com/plug/index.html

CULTe (Le Club des Utilisateurs de Linux Toulousains).http://savage.iut-blagnac.fr/

APODELINE (Paris) http://www.chez.com/juvigny/Asso/

Parinux (Paris) http://www.teaser.fr/~mmestre/parinux.html

Flammekueche Connection (Alsace) http://tux.u-strasbg.fr

SALUG (Sophia Antipolis Linux Users Group)
http://home.dev.amadeus.net/~salug/

Association FINIX (Bretagne).http://www.cyber-espace.com/finix/

Linux Belgian Team :(Liège) http://www.rtfm.be/linux/lbt.html

Linux Québec http://www.linux-quebec.org/

Standardisation

Linux Standard Base (LSB) http://linux.kernel.org/lsb/

Projet de standardisation de Linux soutenu par Linux International.

Sociétés

La liste Pros de Nat Makarévitch répertorie une partie des sociétés de service Linux en France.
http://www.linux-france.com/article/pro/

Les sociétés suivantes commercialisent des distributions, des solutions ou du service autour de Linux. Aucune forme de sélection n'a été effectuée pour constituer la liste de ces prestataires. Nous vous conseillons de contacter une association d'utilisateurs de Linux pour obtenir plus de détails sur ces sociétés.

FNAC
http://www.fnac.fr

Hypermarchés Leclerc (parfois)

Surcouf
139 av Daumesnil
75012 Paris
http://www.surcouf.fr/

Eyrolles
57/61, boulevard Saint Germain
75005 Paris
Tel. : 01 44 41 11 74
Fax : 01 44 41 11 45
librairie@eyrolles.com
http://www.eyrolles.com

MCD2 Diffusion
72, quai des Carrières
94220 Charenton le Pont
Tel. : 01 56 29 23 00
Fax : 01 56 29 23 01
mcd2@mcd2-diff.fr
http://www.mcd2-diff.fr

Alcôve
12 place Indira Gandhi
92230 Gennevilliers
Tel. : 01 47 33 82 84
Fax : 01 47 33 76 98
alcove@alcove.fr
http://www.alcove.fr

KHEOPS
1609, Route de St. Bernard
06225 Sophia Antipolis
Tel. : 04 92 13 29 56
Fax : 04 93 73 34 99
jurix@linux-kheops.com

Pick Systems France
40 Avenue Des Terroirs De France
75012 Paris
Tel. : 01 44 74 55 25
Fax : 01 44 74 55 33
http://www.pick-systems.fr

Atlantic Reseau Informatique
116, av de Marinville
94100 Saint- Maur-des-Fausses
Tel. : 01 45 11 14 14
Fax : 01 45 11 14 10
SuSE@atlantic-reseau.com

MNIS
28, rue Mahias
92100 Boulogne
Tel. : 01 41 10 47 00
Fax. : 01 41 10 47 01
info@mnis.fr
http://www.mmis.fr

IKARIOS
22, rue Voltaire
92000 Nanterre
Tel. : 01 47 24 45 31
lindis@ikarios.com
http://www.ikarios.com/

SORBONE Informatique
23 rue Hotel des Postes
06000 NICE
Tel : 04 93 92 03 03
Fax : 04 93 80 75 69

Arkane Media
27, rue du Fossé des treize
67000 Strassbourg
Tel. : 03 88 32 93 29
Fax : 03 88 32 93 29
arkane@arkane-media.net
http://www.arkane-media.net/

SOS Informatique
23, bd de la Marseillaise
68100 Mulhouse
Tel. : 03 89 60 56 05
Fax : 03 89 60 11 56
linux@sos-informatique.com
http://linux.sos-informatique.com/

BDJ Informatique
116 rue Sainte
13007 MARSEILLE

Tel : 04 91 33 90 40
Fax : 04 91 33 44 48

Atacora S.A.R.L.
Z.A. La Touche d'Anais
16560 Anais
Tel. : 05.45.39.96.00
Fax : 05.45.39.96.00
atacora@wanadoo.f
http://www.atacora.com

Fil CD
11 13 Rue de Corbeil
77111 Soignolles en Brie
France
Tel. : 01 64 06 50 01

TECHNOsciences
14 rue Jacques Lelieur
76000 ROUEN
Tel : 02 35 98 02 36
Fax : 02 35 98 02 36

Librairie du Capitole
3, rue des Lois
31000 Toulouse
Tel. : 05 61 22 51 97
Fax : 05 61 21 73 94
libcap@mipnet.fr
http://www.mipnet.fr/

Forum Espace Culture
50, Grand Place
38000 Grenoble
Tel. : 04 76 22 06 95
Fax : 04 76 22 06 79

Pacific
34, rue Jacques Ferny
76760 Yerville
Tel. : 02 355 680 72
Fax : 02 355 606 24
pdemeure@planete.net

Mandala International
38, Rue Marcel-Dassault
92100 Boulogne-Billancourt
Tel. : 01 47 61 84 90
FAX : 01 47 61 84 91
jclaude@mandalux.com
http://www.mandalux.com/

JC les Pleyades
1 rue Pasqualini, Boite Postale 112
06802 Cagnes su Mer, Cedex
Tel. : 04 93 14 01 55

Caldera Graphics
BP 94, 25 Blvd Wilson
67067 Strasbourg Cedex
Tel. :03 88 21 00 00
FAX : 03 88 75 62 42
mergui@caldera.fr

FRANCE CD
9, rue du chemin de fer
77401 LAGNY SUR MARNE
Tel : 01 64 02 20 02

Sauramps (Librairie)
Le Triangle
34967 MONTPELLIER CEDEX 2
Tel : 04 67 06 78 78 (Poste 313) Stand Informatique
Fax : 04 67 58 27 36

POWER INSIDE
185 rue Charenton
75012 PARIS
Tel : 01.49.28.05.30
Fax :01.49.28.05.33
powerinside@chez.com
http://www.chez.com/powerinside/

Locabureau
126 voie Liberté
57160 Scy Chazelles
Tel : 03 87 60 03 26
http://www.locabureau.com
Ventesinfo@locabureau.com

Medasys Digital Systems
Espace Technologique de St Aubin
Bâtiment « Le Mercury «
91193 Gif sur Yvette Cede
Tel : 01 69 33 73 61
Tel : 01 69 33 73 01
http://www.medasys-digital-systems.fr/linux/
support@medasys-digital-systems.fr

ESC Direct
http://www.escdirect.fr/

Libraire Le Monde en tique
6 rue du Maître Albert

75005 PARIS
Tel : +33-1-55-42-73-73
Fax : +33-1-55-42-73-74
http://www.lmet.fr/
lmet@lmet.fr

Infothèque
32, rue de Moscou
75008 Paris
Tel : 01 45 22 67 01
Fax : 01 42 93 73 83
http://www.infotheque.com/

ATRID Systèmes
2, rue Jean Rostand
91893 ORSAY CEDEX
Tél. 01 69 85 24 13
Fax 01 69 35 50 51
http://www.atrid.fr
infos@atrid.fr

NetWave Erian Concept
Le « Stella » - C/O Guilhem de Wailly
155 bd de la Madeleine
06000 - Nice
Tel : 04 93 44 18 06
Fax : 04 93 14 36 75
http://www.linux-kheops.com/netwave/index.htm
erian@linux-kheops.com

THOT-M INFORMATIQUE,
19 rue Haute
55400 HERMEVILLE
Tel. : 03 29 88 37 74
Fax : 03 29 88 30 56
http://perso.wanadoo.fr/thot-m/
thot-m@wanadoo.fr

Energy Computer
26 rue du Fbg Poissonnière
75010 PARIS
Tél. : 01-48-74-63-12
Fax : 01-49-95-99-16
http://www.energy-computer.com/
pcline@club-internet.fr

Anande
M. Stanis Humez
8, rue Salvador Allende
62200 Boulogne-sur-Mer
Tél : 03.21.83.87.53
Fax : 03.21.87.69.35
http://www.anande.com/
shumez@anande.com

Easter Eggs
1 chemin herbu
78117 Châteaufort
Tél : 33 01 39 56 00 92
Fax : 33 01 39 56 99 61
Informations générales
http://www.easter-eggs.com
info@easter-eggs.com

ALIACOM
20 Rue Hermes
Parc technologique du Canal
31520 Ramonville Saint-Agne France
Tél. : 05 62 19 24 91
Fax : 05 62 19 24 92
http://www.aliacom.fr/
Pierre Baudracco<info@aliacom.fr>

Systèmes & Technologies Informatiques du Ponant
27, rue Auguste Brizeux
22200 Guingamp
Tél : 02 96 43 87 87
Fax : 02 96 43 87 88
http://www.stip.fr
info@stip.fr

Eridan
8, rue des Oziers
ZI du Vert Galan
95310 Saint-Ouen-l'Aumône
Tél : 01 34 64 50 00
Fax : 01 34 30 23 89
http://www.eridan-info.fr
eridan@eridan-info.fr

EBC Consulting, Éric Binger
32, rue au Crampa
La Grange Aux Bois
F-57070 Metz-Grigy
Tél. : 03 87 37 26 83
Fax : 03 87 37 78 30
RNIS : 03 87 37 78 31
http://www.ebcip.fr/ebconsul/
ebconsul@ebcip.fr

Annexe C
Solutions à base Linux

Les systèmes d'exploitation libres mettent sur un pied d'égalité les auteurs et les éditeurs de logiciels commerciaux dans la mesure où tous ont accès aux mêmes outils de développement et au même code source. Il n'est donc pas étonnant qu'un nombre très important de développements libres ou commerciaux soient réalisés en Europe et deviennent des références mondiales. La distribution allemande SuSE est par exemple devenue l'une des rares distributions à être soutenue par la plupart des éditeurs de bases de données dont Oracle. C'est aussi l'une des distributions les plus riches, combinant le meilleur des logiciels libres et des logiciels propriétaires pour Linux.

Le site Europe-Inside (http://www.europe-inside.com) décrit des solutions intégrées faisant appel le plus possible à des logiciels libres ou européens. Ces solutions sont testées et validées dans un cadre professionnel souvent très exigeant. Nous les présentons ici sous une forme abrégée et légèrement adaptée. Par soucis de simplicité, toutes les solutions sont présentés en prenant comme référence la liste des logiciels inclus dans la distribution SuSE 6.0. Dans la mesure où cette partie est libre de droits (voir la licence *Open Content* cet ouvrage), le lecteur est libre de l'adapter et de la publier en prenant n'importe quelle autre distribution comme référence.

Solution 1 : poste bureautique d'entreprise

Linux permet de réaliser un poste bureautique couvrant plus de 95% des besoins informatiques individuels des employés d'une entreprise tous en assurant une excellente compatibilité avec les environnement existants (Windows, MacOS, AS/400, Java, Unix, *mainframe* etc.). Nous suggérons d'installer une distribution SuSE 6.0 (ou mieux), la suite bureautique Star Office, le logiciel d'assistance VNC et le client ICA de Citrix.

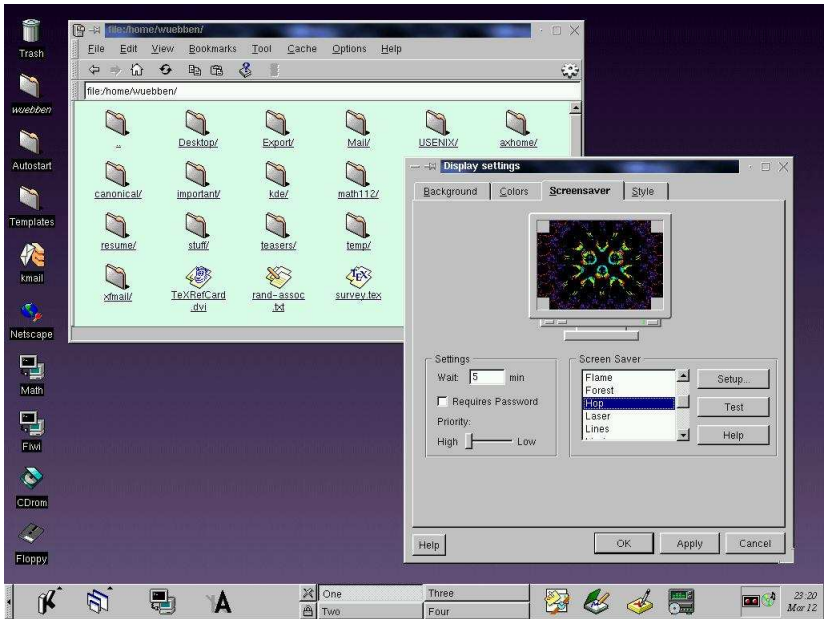

Figure 10. Environnement graphique KDE, gestionnaire de fichiers, configuration de l'écran

La distribution SuSE présente l'avantage d'intégrer le gestionnaire graphique KDE et tous les logiciels nécessaires au partage des fichiers en réseau (Novell, AppleShare, Microsoft SMB etc.). De nombreux logiciels commerciaux très utiles tels que Netscape Communicator, Acrobat Reader ou ssh sont fournis en standard ce qui limite d'autant le recours au téléchargement sur Internet.

Le logiciel propriétaire Star Office assure toutes les fonctions bureautiques de base et offre une interface personnalisable facilitant la transition depuis Windows ou MacOS. Star Office est compatible avec les standards du marché dont Microsoft Office 97. Star Office existe sur d'autres plates-formes (Windows, OS/2, MacOS, Solaris) ce qui facilite l'intégration dans un environnement hétérogène.

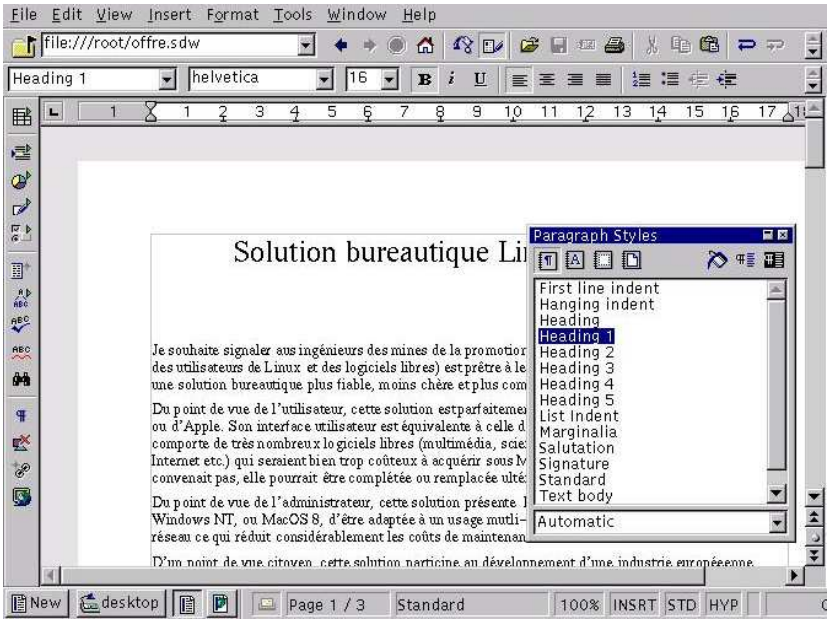

Figure 11. Traitement de texte avec Star Office

Star Office intègre un traitement de texte, un tableur, un logiciel de présentation, un logiciel d'illustration, un logiciel de retouche d'images, un gestionnaire de bases de données relationnelles et un environnement graphique de programmation en BASIC ou en JavaScript. Star Office gère aussi les fonctions de base d'un logiciel de *groupware* adapté au travail en Intranet. Il permet la rédaction de documents HTML, la navigation sur le Web, la gestion du courrier électronique (POP, IMAP, SMTP, LDAP), la gestion d'un agenda en réseau et de groupes de discussion.

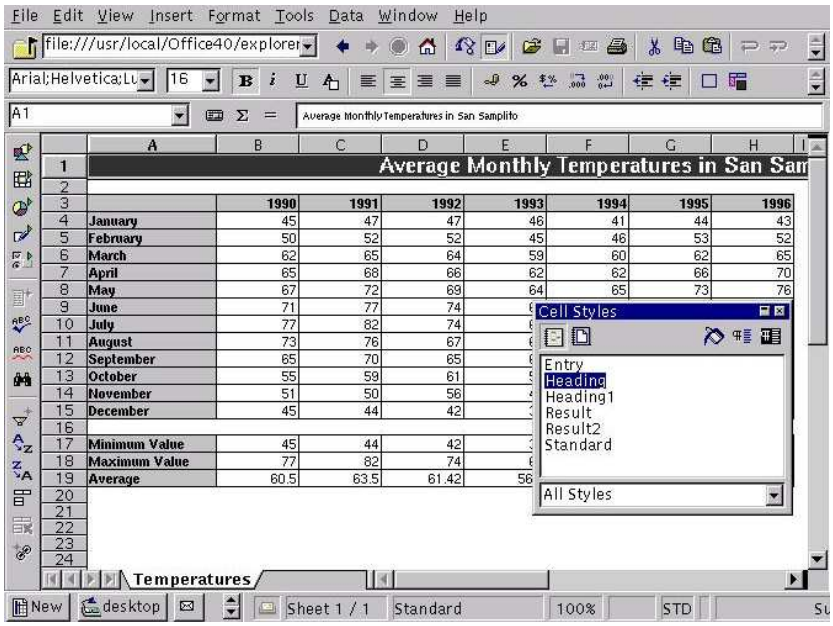

Figure 12. Tableur programmable avec Star Office

Figure 13. Présentation multimédia avec Star Office

Figure 14. Navigateur Web avec Star Office

Contrairement à ce que l'on pourrait croire, un poste bureautique construit autour de Linux est compatible avec les environnements existants, notamment les serveurs de fichiers (Novell, AppleShare, Windows NT, Unix), les applications développées en Java, les applications historiques sous DOS, Windows 3.1 (émulateur DosEmu) ou sous Windows 95/98/NT (accès par le client ICA à un serveur Citrix), les *mainframe* (nombreux émulateurs de terminaux) ou encore les réseaux privées virtuels (protocoles ssh ou IPSEC).

Mieux, ce poste bureautique peut être géré à distance grâce aux protocoles X11R6, aux logiciels VNC et telnet ainsi qu'aux nombreux langages de script (Perl, Tcl etc.), aux shells (sh, csh etc.) et à l'outil de configuration automatique yast. Le logiciel VNC permet notamment d'aider à distance un utilisateur dans un environnement hétérogène (Linux, MacOS, Windows).

Toutes ces caractéristiques, simples mais avancées, peuvent être four-

nies en standard dans un micro-ordinateur puissant de moins de 5000 F. Il faudrait probablement dépenser quatre fois plus pour obtenir une configuration équivalente au moyen de logiciels propriétaires pour MacOS ou Windows.

Tableau 5. Poste bureautique

Produit	*Fonction*	*Type*	*SuSE*
Star Office	Bureautique et *Groupware*	Propriétaire	Oui
Acrobat Reader	Consultation de documents PDF	Freeware	Oui
VNC	Assistance à distance	Libre	Oui
Citrix ICA	Accès aux applications historiques sous Windows	Freeware	Non

Solution 2 : Terminal X

Une version compacte de Linux permet de transformer un micro-ordinateur peu puissant en émulateur de terminal X. La distribution Xdenu tient par exemple sur une ou deux disquettes et transforme en quelques minutes un vieux 386 en terminal X pour accéder à des applications hébergées sur un serveur Linux plus puissant. Ce type de solution permet à des écoles de récupérer des vieux ordinateurs pour offrir à chaque écolier un accès à des applications de type Internet hébergées sur un serveur partagé. Une solution encore plus astucieuse consiste à placer sur une disquette les seuls composants logiciels nécessaires au téléchargement du logiciel d'émulation de terminal X à partir d'un serveur.

Tableau 6. Terminal X

Produit	*Fonction*	*Type*	*SuSE*
XDenu	Emulateur de terminal X	Libre	Non

Solution 3 : *Network Computer*

En partant du principe du Terminal X et en ajoutant un client ICA pour l'accès aux applications Windows ainsi qu'un navigateur voire une machine virtuelle Java, on peut assembler un *Network Computer* très complet. La société parisienne Eridan propose ce type de produit en y intégrant un lecteur de cartes à puce pour gérer les préférences des utilisateurs. Le principal intérêt des *Network Computer* est leur très faible coût de maintenance et leur durabilité puisqu'il suffit de changer le serveur pour augmenter la puissance offerte aux utilisateurs.

Figure 15. Ordinateur de réseau Eridan

Notons que si le produit proposé par Eridan fait appel à des processeur compatibles x86, ce type de *Network Computer* pourrait être réalisé sur une base matérielle quelconque et utiliser, par exemple, des processeurs européens de chez ARM (cas des Netwinder de Corel) ou ST Micro-électronique. La disponibilité du code source des logiciels libres permet d'éviter les phénomènes d'entente entre éditeurs de logiciels et producteurs de micro-processeurs et garantit donc une saine concurrence au niveau du matériel.

Tableau 7. Network Computer

Produit	*Fonction*	*Type*	*SuSE*
XFree	Accès aux applications hébergées sur un serveur Linux ou Unix	Libre	Oui
tn3270	Emulateur de terminal pour accès aux *mainframe*	Libre	Oui
telnet	Emulateur de terminal pour accès aux *mainframe*	Libre	Oui
Communicator	Client pour les applications de *groupware*	Libre	Oui
Citrix ICA	Accès aux applications historiques sous Windows	Freeware	Non
Kaffe option	Machine virtuelle Java	Libre	Oui

Solution 4 : station de création multimédia

Les créateurs multimédia utilisent majoritairement le système MacOS en raison de la qualité et de la productivité des outils de création commerciaux disponibles sur cette plate-forme propriétaire. Mais ces outils logiciels ont un prix élevé ce qui les réserve à des professionnels ou à des entreprises établies.

Pour les autres créateurs, plus nombreux, qui souhaitent disposer d'un environnement de création performant, Linux offre une solution originale disposant d'atouts notables dans le domaine de la 3D, de l'image et du son et de faiblesses temporaires dans le domaine de la vidéo.

Le logiciel libre Gimp est l'un des plus beaux logiciels libres développés à ce jour. Il permet de retoucher ou de créer des images. Il intègre un mécanisme astucieux de modules et de macros qui a conduit des centaines d'utilisateurs à développer de nouvelles fonctionnalités souvent inédites dans les outils commerciaux professionnels comme Adobe Photoshop.

Blender est un environnement de rendu et d'animation 3D développé en interne par une société néerlandaise de production d'images de synthèse qui a souhaité assurer une large diffusion à ses outils sans pour autant devenir éditeur de logiciel. Moonlight Creator permet de compléter Blender pour disposer d'un outil de modélisation libre, et donc adapta-

ble.

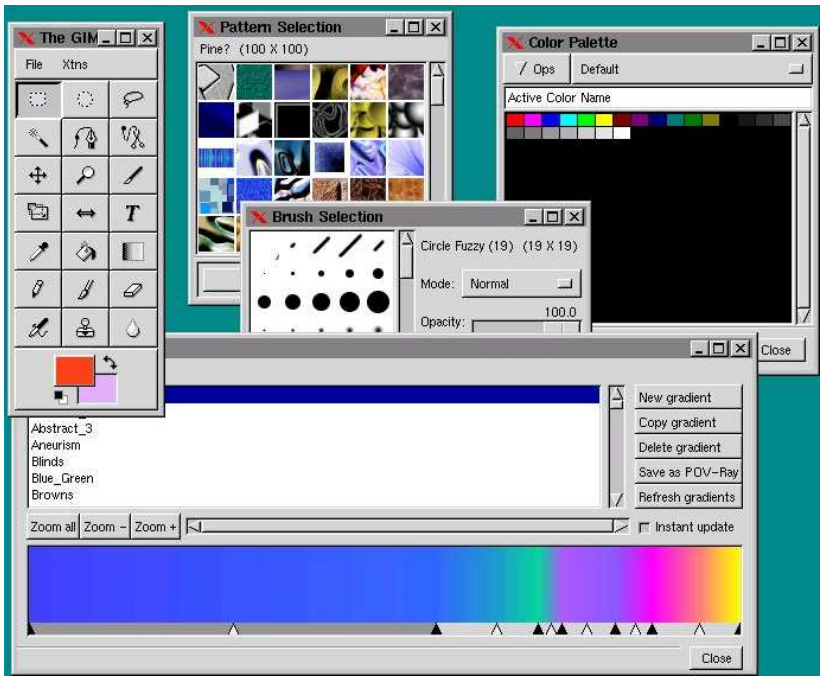

Figure 16. Retouche d'images avec Gimp

Le *freeware* The Sound Laboratory permet de transformer un PC sous Linux en studio d'enregistrement audionumérique 64 pistes. Il est doté, à l'instar de Gimp, d'un système de modules ouvert pour les effets de traitement du son. Les amateurs de MIDI lui préféreront probablement Jazz++, un séquenceur combinant les pistes MIDI à des pistes audio.

Les réalisations multimédias pourront être intégrées sous formes pages Web grâce aux éditeurs HTML fournis avec Netscape Communicator et Star Office ou, pour ceux qui souhaitent graver et diffuser des CD-ROM interactifs et multi-plate-forme (Linux, Unix, MacOS, Windows), assemblées avec le système commercial d'écriture multimédia MetaCard, dont l'usage s'apparente à celui d'Hypercard.

Tableau 8. Station de création multimédia

Produit	*Fonction*	*Type*	*SuSE*
Open Sound System	Gestionnaire de son	Shareware	Oui
Star Office	Bureautiqueet illustration	Propriétaire	Oui
Gimp	Retouche d'images	Libre	Oui
Blender	Synthèse et animation 3D	Freeware	Oui
Moonlight Creator	Modeleur 3D	Libre	Non
The Sound Laboratory	Montage audionomérique 64 pistes	Freeware	Oui
Jazz ou Jazz++	Séquenceur MIDI audionumérique	Shareware	Oui
Netscape Communicator	Editeur HTML	Libre	Oui
Metacard option	Environnement d'écriture multimédia	Propriétaire	Non
XCDRoast	Gravure de CD enregistrables	Libre	Oui

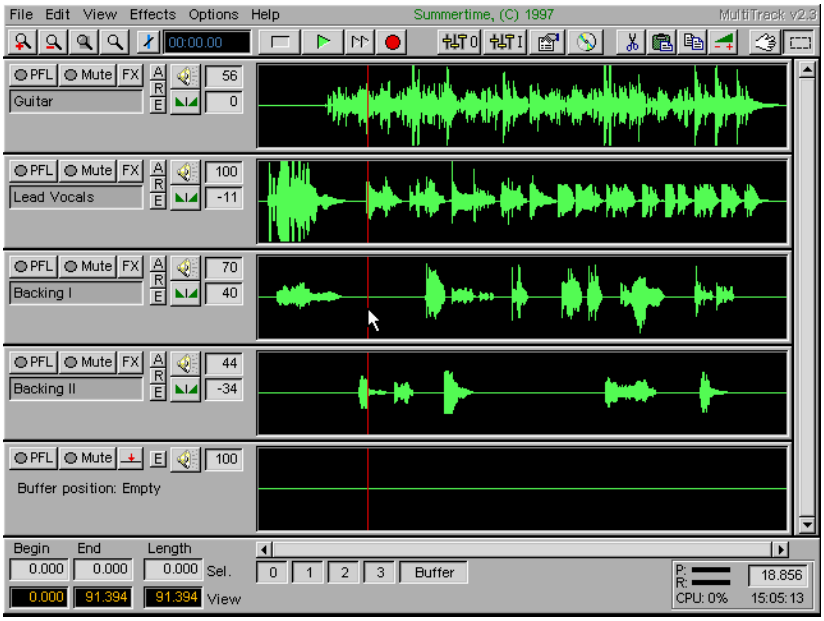

Figure 17. Montage audionumérique avec Multitrack

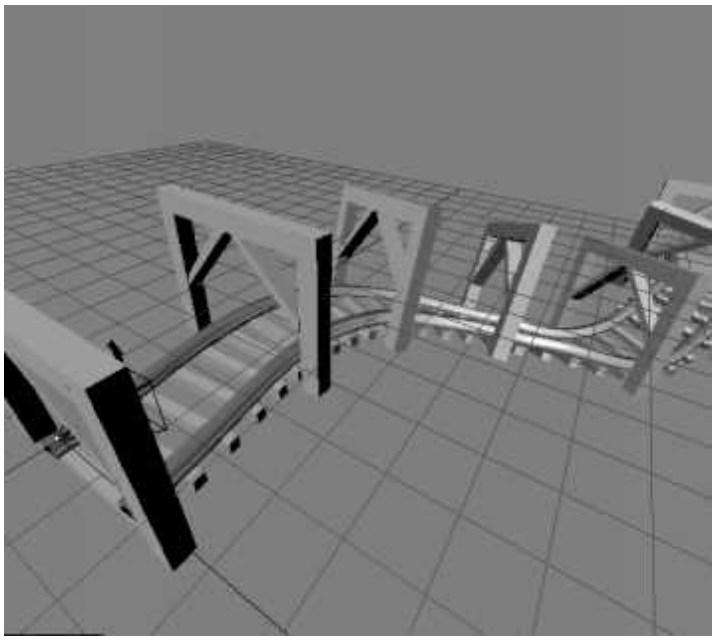

Figure 18. Modélisation 3D avec Blender

Solution 5 : station de travail scientifique

Linux est un excellent choix pour les scientifiques en raison de la profusion d'applications développées par les universités pour les stations de travail Unix et disponibles avec leur code source dans des domaines aussi variés que les mathématiques, les statistiques, la chimie, la physique, la biologie etc. Le fait que Linux tourne sur des ordinateurs à processeurs rapides (PowerPC, Alpha) ou multiprocesseurs permet aux scientifiques de bénéficier de la machine la plus rapide du marché au meilleur prix. Mieux, des systèmes de *clustering* permettent de transformer un réseau de stations Linux en supercalculateur.

Linux permet de fédérer ainsi la puissance de calcul de plusieurs stations sous Linux, grâce à une mise en grappe de ces ordinateurs. Cette solution rivalise avec un supercalculateur Cray, pour un coût sans com-

mune mesure. La solution libre la plus connue est Beowulf, développée à l'origine pour la Nasa, est intégrée par défaut à SuSE et coûte. Elle $30, en combinaison avec Red Hat, dans une distribution connue sous le nom d'Extreme Linux[118].

Cet usage est l'un de ceux qui ont le plus contribué à la célébrité de Linux, avec l'exemple du film « Titanic ». Pour réaliser les images 3D de cette production hollywoodienne, 100 stations Linux Alpha ont été utilisées pour effectuer ensemble les calculs graphiques. Cela a permis d'économiser le prix des licences de l'Unix de Silicon Graphics, Irix, et de diviser par trois le temps de réalisation . « Cette solution a surtout servi pour les animations aquatiques, indique Valérie Delahaye, qui a dirigé une partie du projet pour Digital Domain[119]. Mais NT et Silicon Graphics ont aussi été utilisés. »

Le caractère multitâche d'Unix permet à Linux de réutiliser un réseau informatique pour les calculs. Ainsi, le réseau du Lycée Carnot de Dijon effectue, la nuit, des calculs de factorisation pour l'INRIA.

Pour un usage quotidien, nous recommandons l'environnement de mise en page LateX et son interface graphique KlyX. Le logiciel Scilab, développé par l'INRIA, permet de tester des modèles analytiques alors que MuPAD se spécialise sur les problèmes algébriques. Des outils classiques comme Gimp ou GNU Plot seront aussi très utiles pour préparer les figures des articles de recherche.

118 Pour une critique de ce produit, voir Infoworld du 27 juillet 1998.
119 Entretien avec l'auteur à Montréal (Canada) le 2 décembre 1998.

Tableau 9. Station de PAO scientifique

Produit	*Fonction*	*Type*	*SuSE*
TeX/Latex	Langage de mise en page de documents scientifiques	Libre	Oui
KLyX	Editeur graphique de documents scientifiques	Libre	Oui
Gimp	Visualisation et retouche d'images	Libre	Oui
Scilab	Environnement intégré de calcul scientifique	Libre	Oui
MuPAD	Environnement intégré de calcul formel	Freeware	Oui
GNU Plot	Traceur de courbes	Libre	Oui
Beowulf	Bibliothèque de calcul parallèle	Libre	Oui

Solution 6 : station de développement

La productivité d'un développeur C ou C++ dépend principalement de la rapidité du compilateur, de la qualité de l'interface utilisateur et de la puissance des outils fournis. Linux permet de bénéficier d'un environnement optimal selon ces 3 points de vue. Le fait que Linux tourne sur des ordinateurs à processeurs rapides (PowerPC, Alpha) ou multiprocesseurs permet de bénéficier de la machine la plus rapide du marché. Le débogueur graphique DDD offre une interface utilisateur exceptionnelle d'efficacité. L'éditeur de texte Emacs permet de gérer sous un environnement unique les subtilités de la plupart des langages de programmation.

Les compilateurs GNU ou Cygnus qui sont livrés avec Linux autorisent la compilation croisée et sont disponibles dans de nombreux environnements (Windows, MacOS, Psion, Sony PlayStation etc.). Il est donc tout à fait possible de développer sous Linux une application destinée à une autre plate-forme. La société Metrowerks, spécialiste des environnement de développement multi-plate-forme, annonce d'ailleurs le portage de son environnement Codewarrior pour Linux (processeur PowerPC ou compatible x86).

Restent à choisir les bibliothèques et *frameworks*. Pour les applications généralistes, nous recommandons les bibliothèques WxWindows ou Qt.

WxWindows a l'avantage d'être disponible sur presque toutes les plates-formes, y compris MacOS, mais elle est d'une technologie objet traditionnelle. Qt bénéficie d'une technologie objet séduisante et d'outils de développement rapide. Qt a par exemple permis de développer en un temps record l'environnement graphique KDE. Qt est disponible pour Linux, FreeBSD, les Unix propriétaires et Windows mais pas pour MacOS.

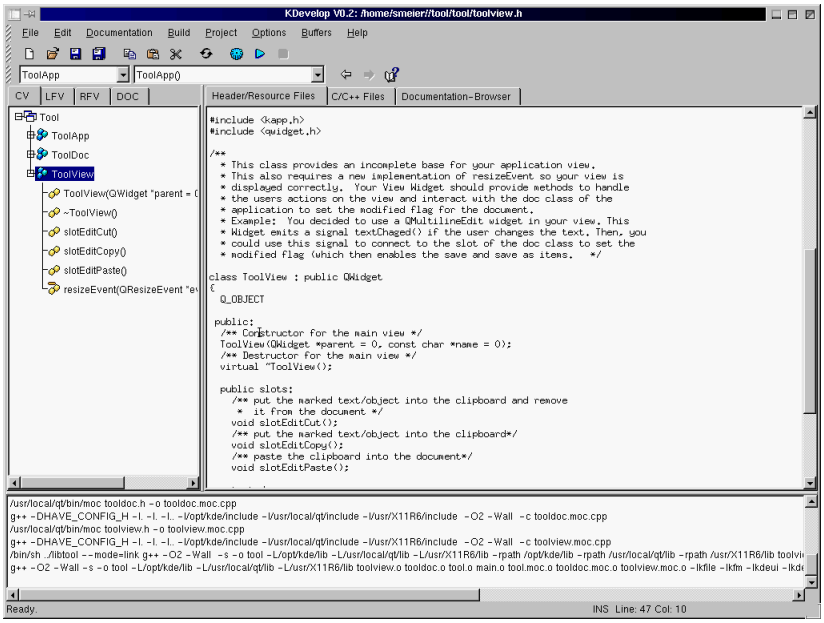

Figure 19. Environnement intégré de développement KDevelop (KDE)

Le développement d'applications 3D est simplifié grâce à la bibliothèque MESA, disponible pour la plupart des systèmes d'exploitation et compatible avec le standard du marché OpenGL. Les applications sonores peuvent être construite autour de l'Open Soud System, seul gestionnaire de son disponible sur tous les Unix. Si la compatibilité avec Windows est nécessaire, en particulier pour développer des jeux vidéos, on peut par exemple recourir à la bibliothèque SDL qui offre un accès stan-

dardisé de bas niveau aux couches graphiques et sonores.

*Tableau 10. Station de développement multi-plate-forme (**L**inux, **W**indows, **M**acOS)*

Produit	*Fonction*	*Type*	*SuSE*	*OS*
GNU	Outils de développement multi-plate-forme	Libre	Oui	LWM
GNU Pro option	Outils de développement multi-plate-forme	Libre	Oui	LW
GNU Emacs	Editeur de texte	Libre	Oui	LW
DDD	Débogueur graphique	Libre	Oui	L
CodeCrusader	Environnement de développement intégré option	Libre	Oui	L
Mesa	Bibliothèque 3D compatible OpenGL	Libre	Oui	LWM
OpenSound System	Gestion du son	Freeware	Oui	L
SDL	Bibliothèque multimédia de bas niveau pour la programmation de jeux vidéos	Libre	Non	LW
Qt	Framework objet avancé	Libre	Oui	LW
wxWindows	Framework	Libre	Oui	LWM
PVM/Beowulf	Bibliothèque de messages pour la programmation concurrente	Libre	Oui	L
JDK	Environnement de développement Java	Semi-libre	Oui	LWM
Kaffe	Machine virtuelle Java	Libre	Oui	LW
NetBeans	Environnement de développement intégré pour Java	Propriétaire	Oui	LWM
Codewarrior option	Environnement de développement intégré multi-plate-forme et adapté aux systèmes embarqués	Propriétaire	Non	LWM
Perl	Langage de script multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL, des outils de traduction de fichiers etc.	Libre	Oui	LWM
Python	Langage à objets multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL et des applications graphiques simples	Libre	Oui	LWM
Tcl/Tk	Langage de script multi-plate-forme et proche du C permettant de développer rapidement des applications graphiques simples	Libre	Oui	LWM
PHP	Langage de script permettant d'intégrer des requêtes SQL au sein d'une page HTML sans lancer de CGI	Libre	Oui	LWM

Bien entendu, Linux permet de développer en Java. Le JDK de Sun est disponible pour la plupart des plates-formes. Metrowerks a publié récemment sa technologie de compilation juste à temps. Netbeans, un logi-

ciel développé par une *start up* à Prague, propose l'un des meilleurs environnements de développement Java compatible JavaBeans. Enfin, Kaffe offre une technologie *Open Source* pour l'exécution des applications Java.

Il ne faut pas oublier non plus les langages de script séduisants disponibles pour Linux : Perl, Python, Tcl, etc.. Ces langages peuvent être très utiles pour un développement rapide ou pour la réalisation d'applications faisant appel à des bases de données. Visual Tcl offre par exemple sous Linux, MacOS et Windows un environnement de développement rapide d'applications clientes similaire à Microsoft Visual Basic.

Solution 7 : *firewall*

La pile TCP/IP intègre des fonctionnalités de *firewall* et de *masquerading* qui ont été utilisées par de nombreuses société pour proposer des solutions intégrées (matériel + logiciel).

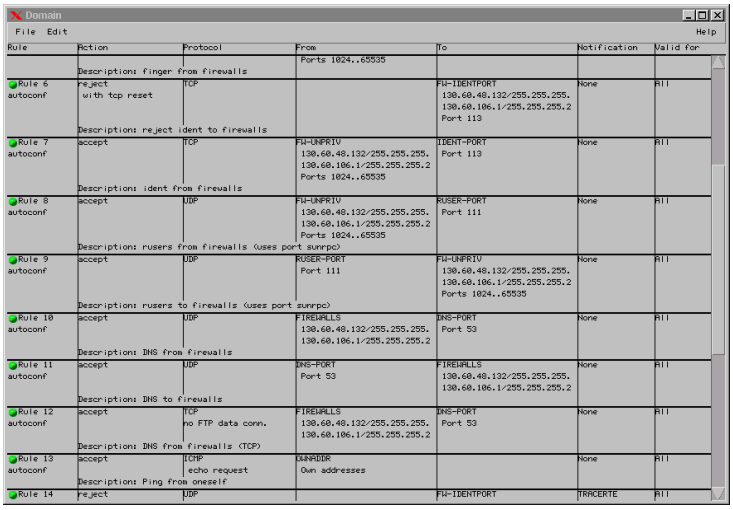

Figure 20. Configuration interactive d'un firewall (Sinus Firewall)

La société CXR[120] développe et produit en France une gamme de routeurs et *firewall* autour de Linux. Le modèle NetLine 101, qui intègre un adaptateur numéris, un routeur et un *firewall*, permet à une PME de se connecter à Internet de façon sécurisée pour moins de 3000F en utilisant un abonnement numéris standard comme ceux proposés par Wanadoo, Magic, Claranet, etc.

La société Locabureau[121] propose le PPX, une solution de *firewall* associée à un serveur de proxy HTTP dans une présentation en rack industriel 19 pouces. Le serveur de proxy, géré par le logiciel libre Squid, permet d'optimiser les transferts en évitant qu'une même page sur le Web soit téléchargée plusieurs fois alors qu'elle pourrait être conservée dans une mémoire cache.

Tableau 11. Firewall

Produit	*Fonction*	*Type*	*SuSE*
Firewall	Scripts de configuration de Linux en *firewall* avec *masquerading*	Libre	Oui
Squid	Serveur de proxy HTTP	Libre	Oui
Firewall How-To	Manuel de configuration de Linux en *firewall*	Libre	Oui
IP-Masquerading Mini How-To	Manuel de configuration de Linux avec *masquerading*	Libre	Oui
Sinus firewall option	Interface graphique pour la configuration du *firewall*	Libre	Non

Solution 8 : réseau privé virtuel

Les meilleurs produits de chiffrement sont probablement développés en Finlande à l'université technologique d'Helsinki. Ils sont souvent disponibles avec leur code source et commercialisés par des sociétés tels que Datafellows ou SSH Inc. Le logiciel ssh (168 bits), et sa version francisée ssf[122] (40 bits), permet de créer une liaison sécurisée et chiffrée

120 http://www.cxr.fr
121 http://www.locabureau.com
122 Cette version est devenue inutile depuis l'adoption de mesures visant à libéraliser l'usage du

entre deux ordinateurs connectés à Internet. En faisant transiter toutes les informations par cette liaison, appelée tunnel chiffré, on peut relier les réseaux locaux d'une entreprise en passant par Internet au prix d'une communication locale et se passer le plus souvent de services d'Intranet plus onéreux reposant sur des liaisons louées ou dédiées. La technique de tunnel chiffré revient à « faire rouler des camions blindés sur les autoroutes » au lieu de « construire des routes grillagées ».

L'usage du chiffrement fort permet de garantir une confidentialité nettement supérieure à celle offerte en pratique par les opérateurs de télécommunication dont les communications sont surveillées par des systèmes d'espionnage électronique tels que le réseau Echelon »[123].

Tableau 12. Réseau privé virtuel chiffré

Produit	*Fonction*	*Type*	*SuSE*
ssh	Logiciel de chiffrement livré avec son code source	Libre	Oui
ssh2 option	Logiciel de chiffrement livré avec son code source	Semi-libre	Non
Linux VPN How-To	Manuel de configuration de Linux en VPN	Libre	Oui
VPS option	Outil de configuration automatique d'un VPN	Libre	Non
F-Secure VPN+ option	VPN compatible IPSEC	Propriétaire	Non

Seules les distributions Linux conçues en Europe peuvent inclure des produits de chiffrement forts exportables dans le reste du monde. La distribution SuSE inclut donc ces produits ainsi que la documentation pour réaliser un réseau privé virtuel. Des produits commerciaux préinstallés sont aussi disponibles, notamment chez Datafellows.

Solution 9 : serveur d'Intranet

Linux a connu un très grand succès auprès des hébergeurs de sites

chiffrement en France.

123 Comment l'Amérique nous espionne. Vincent Jauvert - Nouvel Observateur - N°1779

Web. C'est normal puisque la plupart des outils de l'Intranet sont disponibles sous forme de logiciel libre pour Linux.

Le serveur Apache est l'un des meilleurs serveurs Web du moment. Des modules accélèrent l'exécution des scripts en Perl ou PHP. Des extensions de chiffrement fort sont disponibles et téléchargeables en Europe, ce qui n'est pas le cas de nombreux produits commerciaux américains. La distribution européenne SuSE intègre ces modules et ces extensions.

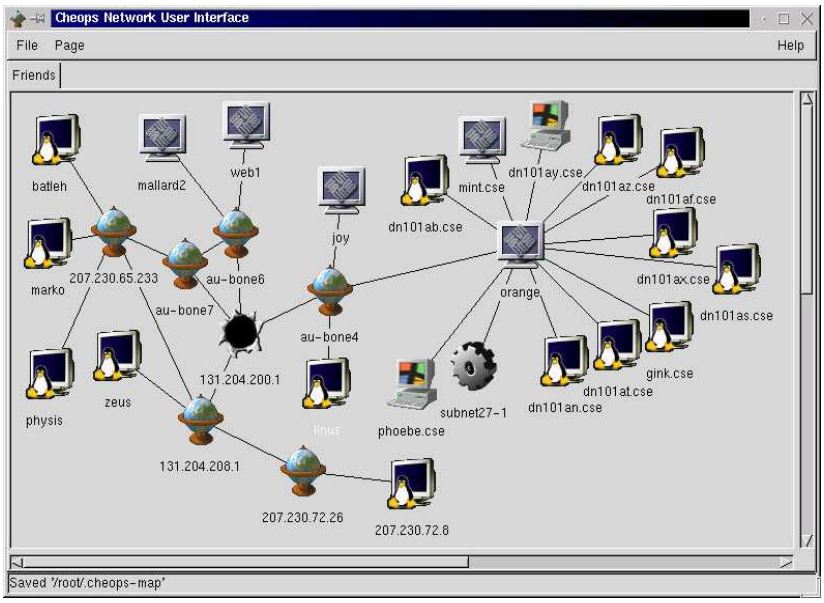

Figure 21. Gestion graphique d'un réseau hétérogène avec Cheops

La gestion du courrier électronique SMTP avec sendmail atteint un niveau de contrôle et de sophistication inégalés. L'une de ses références le plus connues est celle de la Communauté urbaine de Lille, forte d'un parc de 1 200 PC[124]. Celle-ci a remplacé son serveur NT qui intégrait une solution de groupe de travail Lotus Notes par une machine sous Linux et utilisant sendmail, qui s'est avérée plus robuste.

124 Voir Linuxbusiness.com - http://www.linuxbusiness.com/fr/case/cudl.html

Pour ceux qui préfèrent une solution intégrée, la société russe Stalker propose un serveur de courrier électronique que l'on gère à partir d'une interface Web. Le CRU[125] en France a développé un serveur de liste de diffusion, Sympa, qui est utilisé par le ministère de l'éducation nationale et gère correctement les problèmes de caractères accentués contrairement à de nombreux produits commerciaux.

Des fonctionnalités d'Intranet de plus bas niveau peuvent aussi être gérées sous Linux. Le serveur Bind pour la gestion des noms de domaine, un serveur DHCP pour la configuration automatique des adresses IP etc.

Tous les autres protocoles sont bien entendu disponibles sous forme de logiciels libres : groupes de discussion NNTP, annuaires LDAP, serveur de fichiers FTP, serveur de fax en réseau, serveur de messagerie POP ou IMAP. L'ensemble de ces outils peut être combiné grâce à n'importe quel langage de script comme Perl ou Python. Mieux, Cheops est un logiciel libre qui permet de gérer tout un réseau au moyen d'interfaces graphiques

Tous ces services présentés dans le cadre d'un serveur d'Intranet peuvent être utilisés dans le cadre d'applications exigeantes sur Internet. Dejanews, l'un des services d'archivage de *newsgroups* les plus consultés sur Internet, utilise 40 PC Pentium (avec Linux, Apache et Perl) pour gérer les 900 000 messages reçus quotidiennement sur ses forums[126]. La société a choisi cette solution parce qu'elle avait besoin d'une stabilité maximale pour un coût qui ne soit pas celui des Unix propriétaires.

125 http://www.cru.fr
126 « Taming Newsgroups » - Michelle V. Rafter - Infoworld, 2 mars 1998

Tableau 13. Serveur d'Intranet

Produit	*Fonction*	*Type*	*SuSE*
Apache	Serveur Web	Libre	Oui
Sendmail	Serveur de courrier électronique paramétrable	Libre	Oui
Communigate Pro option	Serveur de courrier électronique à interface Web	Libre	Non
Sympa	Gestionnaire avancé de listes de diffusion	Libre	Non
bind	Serveur de nom de domaine	Libre	Oui
dhcp	Serveur DHCP	Libre	Oui
Internet Net News	Serveur de groupes de discussions	Libre	Oui
Open LDAP	Serveur d'annuaires LDAP	Libre	Oui
Wu FTP	Serveur FTP	Libre	Oui
Hylafax	Serveur de fax en réseau	Libre	Oui
pop	Serveur de courrier électronique POP	Libre	Oui
imap	Serveur de courrier électronique IMAP	Libre	Oui
Perl	Langage de script multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL	Libre	Oui
Python	Langage à objets multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL	Libre	Oui
Cheops	Outils graphique d'administration de réseau	Libre	Non

Solution 10 : serveur de fichiers

La plupart des distributions Linux intègrent tous les logiciels nécessaires pour mettre sur pied un serveur de fichiers adapté à un environnement hétérogène (MacOS, Windows, Unix). Samba assure une compatibilité parfaite avec les réseaux Microsoft (norme SMB). Mars remplace avantageusement un serveur de fichiers Novell. Netatalk permet aux Macintosh d'accéder aux fichiers en partage sur un serveur Linux tout en garantissant une conversion des formats de fichiers. Enfin, le serveur NFS livré avec Linux propose un niveau de performance équivalent aux Unix propriétaires.

Bien entendu, les services d'impression en réseau sont aussi gérés. La société Tefal (ustensiles de cuisson) utilise ainsi deux serveurs Linux (des

Pentium 120 avec 8 Mo) pour imprimer ses 250 000 factures mensuelles. Et la librairie Lavoisier[127], spécialisée dans l'édition scientifique, utilise un serveur avec Pentium 150 sous Linux reliés à deux modems pour gérer l'envoi des fax expédiés par ses postes clients.

La sauvegarde est une autre fonctionnalité essentielle sur un serveur. Pour compléter dump, l'outil rustique de gestion de sauvegarde d'Unix, des éditeurs ont conçu des logiciels propriétaires sophistiqués comme Arkeia dont une version de démonstration est livrée avec SuSE.

Tableau 14. Serveur de fichiers

Produit	*Fonction*	*Type*	*SuSE*
Samba	Serveur de fichiers et d'impression compatible SMB	Libre	Oui
Netatalk	Serveur de fichiers et d'impression compatible AppleShare	Libre	Oui
Mars	Serveur de fichiers et d'impression compatible Novell	Libre	Oui
nfs	Serveur de fichiers compatible NFS	Libre	Oui
lpr	Serveur d'impression compatible BSD	Libre	Oui
Hylafax	Serveur de fax en réseau	Libre	Oui
Dump	Logiciel de sauvegarde	Libre	Oui
Arkeia option	Logiciel de sauvegarde	Propriétaire	Oui

Mais, quel que soit le système d'exploitation, l'installation d'un logiciel serveur est toujours une tâche délicate car il est nécessaire de procéder à des tests longs et poussés du matériel pour s'assurer de sa fiabilité. Lorsque l'on ne dispose pas des compétences nécessaires, il vaut alors mieux faire appel à une société de services spécialisée. Autre solution, bien adaptée aux PME : acquérir un micro serveur préinstallé[128]. Cobalt, une *start up* ayant France Telecom et NTT comme actionnaires, offre le Qube, un serveur Risc pour groupe de travail à $999, livré avec des outils libres (Linux, Apache, Sendmail, DNS, Perl, etc.) et, pour certaines versions, des outils de *groupware*. Corel a lui porté Linux sur les processeurs

127 http://www.lavoisier.fr
128 « Computer Reseller News » (édition française), 5 novembre 1998.

Strong ARM et le distribue sur son offre de microserveurs « Netwinder ».

Solution 11 : serveur de base de données

Presque tous les éditeurs de serveurs de bases de données (SGBD) ont porté leurs produits pour Linux en espérant proposer une solution d'entrée de gamme capable de concurrencer Microsoft.

L'offre est si vaste qu'il est difficile de choisir. Nous recommandons PostgreSQL dans deux cas : nécessité d'utiliser un logiciel libre ou besoin de fonctionnalités très avancées. Le principal défaut de Postgres est sa lenteur. MySQL est une excellente solution en termes de fiabilité, de légèreté et de rapidité ce qui contribue à son succès auprès des hébergeurs de sites Web. Son défaut le plus criant est de faire l'impasse sur les transactions. Primebase gère les transactions, est assez bon marché et est disponible sur presque toutes les plates-formes (Linux, Solaris, Windows, MacOS). Il a été intégré par Golive dans un système de *workflow* HTML ce qui devrait lui assurer une base d'utilisateurs importante. Solid a des fonctionnalités équivalentes à PrimeBase mais est développé par une équipe plus importante. Adabas est un serveur haut de gamme dont certaines versions sont agréées SAP[129]. Enfin, O2 est l'une des rares bases de données véritablement objet du marché et est disponible depuis peu pour Linux.

Les langages Perl, Python ou PHP offrent de bonnes fonctionnalités pour développer une application interactive sur le Web faisant appel à une base de donnée. Notons l'existence de FaceMySQL et de Genquery pour développer rapidement ou automatiquement le code d'interface et de consultation d'une table à partir d'un navigateur.

129 SAP a annoncé dans son magazine (http://www.sapmag.de) lors d'un entretien avec Karl-Heinz Hess que R/3 est désormais disponible sous Linux mais pour l'instant sans assistance.

Tableau 15. Serveur de bases de données

Produit	*Fonction*	*Type*	*SuSE*
PostgreSQL	SGBD objet avancé	Libre	Oui
MySQL	SGBD relationnel compact et rapide, livré avec son code source, très utilisé par les hébergeurs de sites Web	Semi-libre	Oui
Adabas option	SGBD relationnel multi-plate-forme haut de gamme	Propriétaire	Oui
Solid option	SGBD relationnel multi-plate-forme	Freeware	Oui
PrimeBase option	SGBD relationnel multi-plate-forme	Freeware	Non
O2 option	SGBD objet multi-plate-forme	Freeware	Oui
Perl	Langage de script multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL	Libre	Oui
Python	Langage à objets multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL	Libre	Oui
PHP	Langage de script permettant d'intégrer des requêtes SQL au sein d'une page HTML sans lancer de CGI	Libre	Oui
FaceMySQL	Script PHP de développement rapide d'interface Web	Libre	Non
genquery	Script Perl de développement rapide d'interface Web	Libre	Non

Solution 12 : hébergement Web

L'hébergement d'un site requiert une configuration intermédiaire entre la solution Intranet et la solution SGBD présentées ci-dessus. Le logiciel VShop permet de réaliser des sites sophistiqués de commerce électronique (vente en ligne, stock, facturation, etc.). La disponibilité d'annuaires LDAP n'est pas encore très fréquente ce qui est regrettable. La principale différence entre hébergeurs est la qualité de l'interface de configuration (fichiers texte, Web) et la qualité des services d'hébergement virtuel (CGI, *mail*, etc.).

Tableau 16. Hébergement Web

Produit	*Fonction*	*Type*	*SuSE*
Apache	Serveur Web	Libre	Oui
Sendmail	Serveur de courrier électronique paramétrable	Libre	Oui
Sympa	Gestionnaire avancé de listes de diffusion	Libre	Non
Open LDAP	Serveur d'annuaires LDAP	Libre	Oui
Wu FTP	Serveur FTP	Libre	Oui
pop	Serveur de courrier électronique POP	Libre	Oui
MySQL	SGBD relationnel compact et rapide, livré avec son code source, très utilisé par les hébergeurs de sites Web	Semi-libre	Oui
Perl	Langage de script multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL	Libre	Oui
PHP	Langage de script permettant d'intégrer des requêtes SQL au sein d'une page HTML sans lancer de CGI	Libre	Oui
Tcl	Langage de script multi-plate-forme et proche du C permettant de développer des CGI	Libre	oui
Python	Langage à objets multi-plate-forme permettant de développer des CGI faisant appel à des requêtes SQL	Libre	Oui
VShop	Solution avancée de commerce électronique	Propriétaire	Oui
Virtual Services HOWTO	Notice de configuration d'une hébergement virtuel (plusieurs IP, plusieurs domaines sur une même machine)	Libre	Oui

Annexe D
Les grands logiciels pour Linux

Cette annexe a été rédigée en coopération avec Linux-Center (http://www.linux-center.org), le site Web bilingue de référence sur les applications pour Linux. Une partie des droits d'auteur de cet ouvrage seront reversés à Linux Center.

On trouvera également de nombreuses applications sur le site http://www.linuxapps.com ainsi que sur le site du projet KDE à la page http://www.kde.org/apphomepages.html

Distributions

Comparaisons

The Linux Distribution HOWTO http://metalab.unc.edu/pub/Linux/docs/HOWTO/Distribution-HOWTO

Ce document est conçu pour aider un utilisateur à choisir une distribution Linux. Il s'intéresse principalement aux distributions sur CD-ROM pour les ordinateurs à architecture compatible Intel.

Choisir une distribution (Nat Makarévitch) http://www.linux.org/help/lists.html

Comparaison de distributions (Red Hat, Debian, Slackware, autres).

Linux distributions compared http://www.ssc.com/lj/distable.html

Comparaison des principales distributions Linux, par l'équipe du Linux Journal.

Distributions généralistes

S.u.S.E. http://www.suse.com/

Distribution multilingue (anglais, allemand, français, espagnol) d'origine allemande comprenant une collection particulièrement complète de logiciels libres et de versions de démonstration de logiciels commerciaux dont Star Office. Intègre des outils de chiffrement et d'optimisation de serveurs Web.

Debian http://www.debian.org/

Projet collectif, commencé sous l'égide de la FSF, d'une distribution de référence exhaustive.

Linux Mandrake http://www.linux-mandrake.com/

Nouvelle distribution d'origine française qui intègre une Red Hat, KDE et des outils facilitant l'usage de Linux par un débutant. Cette distribution semble bien adaptée au poste client. Existe en version française et dans de nombreuses autres langues.

Redhat http://www.redhat.com/

Le leader mondial des distributions Linux. Existe à la fois pour les ordinateurs à architecture compatible Intel, les ordinateurs à processeur Alpha et de nombreuses autres architectures. Existe en version française.

Turbo Linux http://www.pht.com

La distribution leader au Japon. Existe à la fois pour les ordinateurs à architecture compatible Intel et les ordinateurs à processeur PowerPC.

Caldera OpenLinux http://www.calderasystems.com/products/openlinux/index.html

Distribution commerciale « haut de gamme".

LinuxPPC http://www.linuxppc.org

La distribution leader pour les Macintosh et les ordinateurs à processeur PowerPC.

Distributions spécialisées

xdenu http://xdenu.tcm.hut.fi/

Distribution très compacte conçue pour transformer un PC en terminal X.

Trinux http://www.trinux.org

Distributions compacte tenant sur 2 ou 3 disquettes qui permet de transformer un PC en routeur, *firewall* et système de sécurité réseau.

Linux Router Project (LRP) http://www.psychosis.com/linux-router/

Distribution conçue pour tenir sur une disquette et mettre en œuvre des applications embarquées de communication (routeur, serveur de terminal etc.).

rtlinux http://luz.cs.nmt.edu/~rtlinux/

Extensions temps-réel de Linux.

Interface graphique

KDE (The Kool Desktop Evironment) http://www.kde.org

Projet ambitieux d'un environnement de travail graphique complet et convivial pour Linux et autres UN*Xes. KDE n'est pas seulement un système de fenêtrage mais un ensemble complet d'applications et d'utilitaires qui font désormais de Linux un système d'exploitation au moins aussi convivial que les systèmes propriétaires. KDE utilise la librairie Qt.

GNOME (GNU Network Object Model Environment) http://www.gnome.org/

Ce projet a pour but de construire une interface utilisateur complète et facile à utiliser, basée exclusivement sur le logiciel libre.

Fvwm http://fvwm.math.uh.edu/

Le window manager le plus répandu. Configurable à volonté, certaines de ses variantes (par ex : fvwm95 et AfterStep) permettent même d'imiter à la perfection certains systèmes propriétaires.

AfterStep http://www.afterstep.org/ et **Window Maker** http://www.windowmaker.org/

Systèmes de gestion de fenêtres offrant un aspect similaire à l'environnement NeXTStep.

Enlightment http://www.enlightenment.org/

Gestionnaire de fenêtres de conception récente.

X-Plorer http://www.x-plorer.org/

Gestionnaire graphique de fichiers semblable à celui de systèmes propriétaires.

Bureautique

Suite bureautique

Staroffice http://www.stardivision.com/

Suite bureautique commerciale modulaire (traitement de texte, tableur, grapheur, base de données), gratuite pour usage privé et dans l'enseignement. Leader de son marché (sous Windows) en Allemagne. Existe aussi sous MacOS, OS/2, Windows et Solaris.

SIAG Suite
http://www.edu.stockholm.se/siag/

Siag Office est une suite bureautique pour Unix, qui comprend un traitement de texte (PW), un tableur (SIAG) et un logiciel de présentation (EGON).

Applixware
http://www.applix.com/appware/linux/

Suite bureautique commerciale (traitement de texte compatible *Word*, tableur, grapheur, courrier électronique, environnement de travail HTML).

Cliq http://www.quad.com/

Suite bureautique commerciale (traitement de texte modulaire, courrier électronique, base de données).

Axene Office http://www.axene.com

Suite bureautique commerciale complète et peu onéreuse (traitement de texte, tableur, mise en page, dessin etc.).

KOffice http://koffice.kde.org

Suite bureautique libre en développement (versions alpha et bêta) pour KDE comprenant traitement de texte (KWord, KLyx), logiciel de présentation (KPresenter), logiciel de dessin (KIllustrator), tableur (KSpread), retouche d'images (KImage), formules (KFormula) et graphiques (KChart). Utilise un mécanisme de document composite de type OLE ou OpenDOC.

Tableur

Linux Spreadsheets
http://www.hex.net/~cbbrowne/spreadsheets.html

Liste commentée de tableurs pour Linux.

xquad
http://www.axene.com/french/xquad.html

Tableur commercial (licence a prix raisonnable pour éducation, Linux, FreeBSD, NetBSD).

KSpread
http://koffice.kde.org/kspread/index.html

Tableur libre de la suite KOffice.

WingZ http://www.wingz.com/

Tableur *shareware*. Existe aussi sous Windows et MacOS.

XESS http://www.ais.com/

Tableur commercial / *shareware*.

Editeurs de texte

The linux editor documentation project
http://pegasus.rutgers.edu/~elflord/linux/editors/

Comparaison d'éditeurs *simples* (MCedit, EE, Pico, JED, Xenon, Medit, etc.).

XEmacs http://www.xemacs.org/

Clone de l'éditeur Emacs, sous X11 et NT. Facile d'accès, grâce à son système de menus, il est reconfigurable à l'infini, grâce à son langage de scripts (EMACS-LISP). Sauf pour les inconditionnels de *vi*, c'est donc une des pierres angulaires de tout système Un*x.

Vim (Vi IMproved)
http://www.vim.org/

L'un des nombreux clones de l'éditeur original *vi*, probablement le plus réussi. Les dernières versions intègrent une interface graphique (X11), le *syntax highlighting* et sont programmables à l'aide (au choix) des langages de scripts Python.

Nedit http://www-pat.fnal.gov/nirvana/nedit.html

Editeur de texte graphique avec des fonctionnalités équivalentes à emacs.

Traitement de texte

Word Processors for Linux
http://www.hex.net/~cbbrowne/wp.html

Comparaison de traitements de textes sous Linux.

Maxwell
http://www.eeyore-mule.demon.co.uk/

Traitement de texte simple mais efficace pour Linux.

Corel WordPerfect for Linux
http://linux.corel.com/

Portage sous Linux du célèbre traitement de texte commercial.

Axene Xallwrite
http://www.axene.com/english/xallwrite.html

Logiciel de traitement de texte pour Linux et les autres Un*x.

KWord
http://koffice.kde.org/kword/index.html

Logiciel de traitement de texte de la suite KOffice.

Présentation

MagicPoint http://www.Mew.org/mgp/

Excellent logiciel de présentation libre.

kpresenter http://koffice.kde.org/kpresenter/index.html

Logiciel de présentation de KOffice, la suite bureautique de KDE.

Illustration

KIllustrator http://koffice.kde.org/killustrator/index.html

Logiciel libre de dessin vectoriel de la suite KOffice.

Sketch
http://www.online.de/home/sketch/Default.htm

Logiciel libre de dessin vectoriel programmable en Python.

PAO

Axene Xclamation
http://www.axene.com/english/xclamation.html

Axene Xclamation permet de réaliser facilement des documents d'excellente qualité : gestion multi-documents, multi-colonnage et gouttières, règles magné tiques, gestion de cadres polymorphes pour intégration de textes et d'images bitmap ou vectorielles, positionnements et manipulations complexes des cadres (opérations logiques sur cadres, détourage, zoom, rotations, transparence du fond, etc.), exportation de texte au format HTML.

FrameMaker
http://www.adobe.com/prodindex/framemaker/main.html

Logiciel de mise en page adapté aux documentations techniques. Nécessite la bibliothèque IBCS pour fonctionner sous Linux.

Finance

Moneydance http://seanreilly.com/java/moneydance.html

Excellent logiciel libre de finances personnelles, entièrement développé en Java.

Gnucash http://www.gnucash.org/

Gestionnaire de finances personnelles.

Correcteur orthographique

ispell
http://fmg-www.cs.ucla.edu/geoff/ispell.html

Correcteur orthographique interactif (possibilité d'utiliser un dictionnaire français).

SGML/HTML/XML

The SGML/XML Web Page
http://www.oasis-open.org/cover/publicSW.html

Liste exhaustive et à jour de logiciels SGML et XML gratuits (parseurs, formateurs, éditeurs, etc.).

Sgmltools http://www.sgmltools.org/

Logiciel de formatage de documents SGML, qui permet de produire des documents LaTeX, PostScript (via LaTeX), HTML, GNU info, LyX, RTF (W*rd) et ASCII (via groff) à partir d'une seule source SGML. Utilisé entre autre pour produire la documentation de Linux.

Thot
http://www.inrialpes.fr/opera/Thot/index.html

Traitement de texte structuré (SGML), programmable, développé par l'INRIA.

James Clark Tools
http://www.jclark.com/

Outils très performants pour le traitement de documents SGML, XML et HTML.

MSWordView
http://www.csn.ul.ie/~caolan/docs/MSWordView.html

Convertit un document Word en document HTML.

TTH
http://hutchinson.belmont.ma.us/tth/tth.html

Convertisseur Tex/LaTeX vers HTML.

PAO scientifique

teTeX http://www.tug.org/teTeX/

Distribution de référence de TeX pour Linux. Tex/LaTeX est système de mise en page de documents universel, programmable, utilisé principalement dans la communauté scientifique.

LyX http://la1ad.uio.no/lyx/

Traitement de texte WYSIWYG, basé sur LaTeX2e, mais qui ne nécessite pas la connaissance de ce dernier.

KLyX http://www-pu.informatik.uni-tuebingen.de/users/ettrich/klyx/klyx.html

Version de *LyX* pour KDE.

tex2rtf
http://web.ukonline.co.uk/julian.smart/tex2rtf/

Tex2RTF permet de convertir un document LaTex en fichier RTF, en fichier d'aide Windows ou en document HTML.

Clients Communication

Courrier électronique

pine http://www.washington.edu/pine/

mutt http://www.mutt.org/

elm http://www.myxa.com/elm.html

xfmail
http://Burka.NetVision.net.il/xfmail/

Kmail http://studserver.uni-dortmund.de/~su0197/kde/kmail/index.html

Navigateur Web

Netscape Communicator
http://www.mozilla.org.

Amaya http://www.w3.org/Amaya/

Le navigateur expérimental officiel du W3C, basé sur l'éditeur de documents structurés (SGML) de l'INRIA.

Lynx http://lynx.browser.org/

Navigateur Web tout texte, à utiliser en mode terminal.

FTP

ncftp http://www.ncftp.com/

sprocket
http://jungfrau.ptf.hro.nl/sprocket/

News

tin http://www.tin.org/

knews
http://www.matematik.su.se/~kjj/

PerUser http://peruser.netpedia.net/

Internet Relay Chat

sirc http://www.eleves.ens.fr:8080/home/espel/sirc.html

ksirc http://zuel.ksirc.org/

kvirc http://www.kvirc.org/

Fax

Hylafax http://www.hylafax.org/

Faximum http://www.faximum.com/

Minitel

xtel
http://www.alienor.fr/~pierre/xtel/index.html

Serveur Communication

Web

Apache http://www.apache.org

Roxen http://www.roxen.com/

Squid http://squid.nlanr.net/Squid/

Cache HTTP permettant d'optimiser les accès au Web d'un groupe de travail.

Analyse de sites

Webalizer http://www.mrunix.net/webalizer/

Statistiques d'accès à un serveur Web.

Analog
http://www.statslab.cam.ac.uk/~sret1/analog/

Statistiques d'accès à un serveur Web.

Weblint
http://www.cre.canon.co.uk/~neilb//weblint/

Vérificateur de pages HTML. Indispensable.

MOMspider
http://www.ics.uci.edu/pub/websoft/MOMspider/

Programme de vérification de sites Web.

Courrier électronique

Exim http://www.exim.org/

Qmail http://www.qmail.org/

IBM Secure Mailer (*Open Source*)
http://www.alphaWorks.ibm.com/formula/securemailer

Communigate http://www.stalker.com

Sendmail http://www.sendmail.com

Fetchmail
http://www.tuxedo.org/~esr/fetchmail/

Programme de transfert de courrier électronique sur des connexions temporaires SLIP ou PPP. Supportes les standards POP2, POP3, RPOP, APOP, KPOP, IMAP et ESMTP ETRN. Les messages peuvent ensuite être consultés par les outils habituels comme elm ou mail.

procmail
http://www.ii.com/internet/robots/procmail/

Puissant outil de traitement automatisé des messages électroniques.

Spamcan
http://consult.ml.org/~timb/spamcan/

Système anti-spam permettant de filtrer les courriers électroniques indésirables.

Liste de diffusion

Sympa http://listes.cru.fr/sympa/

Majordomo
http://www.greatcircle.com/majordomo/

Fax

Hylafax http://www.hylafax.org/

Faximum http://www.faximum.com/

Groupware

CDC Intrastore
http://intrastore.cdc.com/www/

Solution de *groupware* commerciale, qui utilise les standards de l'Internet.

Infoplace http://www.infoplace.com/

Solution de groupware intégrée notamment dans le Qube de cobalt.

Teamwave Workplace
http://www.teamwave.com/

Logiciel de travail en groupe (*groupware*) pour Un*x et certains OS propriétaires. Commercial, gratuit pour un essai ou pour les enseignants.

BSCW http://bscw.gmd.de/

Logiciel de travail en groupe, écrit en Py-

thon, primé par le « European Software Innovation Award (ESIP '96) », disponible en plusieurs langues, dont le français, et gratuit.

News

inn http://www.isc.org/inn.html

FTP

WU-FTPD
http://www.academ.com/academ/wu-ftpd/

BeroFTPD ftp://beroftpd.unix.eu.org/pub/BeroFTPD/

Proftpd http://www.proftpd.org/

Fichiers et impression

Samba http://www.samba.org/

Serveur de fichiers et d'impression SMB (pour servir des machines sous système d'exploitation propriétaire).

Netatalk
http://www.umich.edu/~rsug/netatalk/ et http://artoo.hitchcock.org/~flowerpt/projects/linux-netatalk/

Implémentation du protocole AppleTalk pour Unix, pour partager des ressources avec les Macintosh.

Mars NWE
ftp://linux01.gwdg.de/pub/ncpfs/

Emulation de serveur Novell.

Caldera Netware for Linux
http://www.calderasystems.com/products/netware/index.html

Annuaire

OpenLDAP http://www.openldap.org/

Réplication FTP et HTTP

wget ftp://ftp.gnu.org/pub/gnu/

w3mir
http://www.math.uio.no/~janl/w3mir/

ECLIPT Mirror http://eclipt.uni-klu.ac.at :8080/projects/emirror/

Moteurs de recherche

Echo http://www.echo.fr

Glimpse
http://glimpse.cs.arizona.edu/webglimpse/index.html

ht://Dig http://www.htdig.org/

Firewall

ipfwadm
http://www.xos.nl/linux/ipfwadm/

Sf firewall
http://www.ifi.unizh.ch/ikm/SINUS/firewall.html

Sécurité

ssh et ssh2 http://www.ssh.fi

Administration sécurisée et tunneling chiffré simple.

VPS http://www.strongcrypto.com/.

Solution de réseau privé virtuel libre.

F-Secure VPN
http://www.datafellows.com/

Solution commerciale de réseau privé virtuel

Commerce électronique

minivend http://www.minivend.com/

perlshop
http://www.arpanet.com/PerlShop/PerlShop.html

vshop http://www.vshop.de/

Services de base

dhcpd http://www.phystech.com/

Serveur DHCP.

BIND http://www.isc.org/

Serveur de noms de domaines

Administration

linuxconf http://www.solucorp.qc.ca/linuxconf/

Cheops http://www.marko.net/cheops/

Gestion graphique très complète d'un réseau.

Webmin http://www.webmin.com/

Administration des services de réseau à travers une interface Web.

Bases de données

Linux SQL Databases and Tools http://linas.org/linux/db.html

Liste exhaustive et commentée de SGBDs sous Linux.

Bases de données libres /semi-libres

PostgreSQL http://www.postgresql.org/

Système de gestion de bases de données relationnelles-objets. Implante l'essentiel du standard SQL, y compris des extensions récentes ou en cours de standardisation (SQL98).

MySQL http://www.mysql.com

Système de gestion de bases de données relationnel (SGBDR). Implante une partie substantielle du standard SQL, mais pas les transactions. *shareware* mais sources disponibles. Elue meilleur SGBD de l'année 1998 par CNET. De nombreux scripts sont fournis dont FaceMySQL qui offre un accès aux données et structures de la base à travers le Web sans devoir programmer.

Bases de données propriétaires

Pick D3 http://www.pick.com

Yard http://www.yard.de/

Primebase http://www.snap.de/Text/PrimeBase/db.html

Adabas http://www.sag.de

SOLID SQL Serveur http://www.solidtech.com/

Informix-SE http://www.informix.com/

Ingres II http://www.cai.com/products/ingres.htm

Empress http://www.empress.com/

Interbase http://www.interbase.com

Oracle http://www.oracle.com

O2 http://www.o2tech.fr/

Base de donnée relationnelle-objets, compatible SQL et ODBC, commerciale.

Virtuoso http://www.openlinksw.com/virtuoso

Base de donnée virtuelle permettant d'unifier les accès à des bases de données hétérogènes.

Sciences et ingénierie

Scientific Applications on Linux http://SAL.KachinaTech.COM/

Index très complet d'applications scientifiques et professionnelles (gratuites, *shareware* ou commerciales) qui tournent sous Linux.

Statistiques

fiasco http://www.fsf.org/software/fiasco/index.html

xldlas http://a42.com/~thor/xldlas/

MacAnova

http://www.stat.umn.edu/~gary/macanova/macanova.home.html

R http://www.ci.tuwien.ac.at/R/contents.html

Calcul formel

MuPAD http://www.mupad.de

Maple http://www.maplesoft.com/

Mathematica http://www.wolfram.com/

Macsyma http://www.macsyma.com/

Magma http://www.maths.usyd.edu.au:8000/u/magma/

Macaulay2 http://www.math.uiuc.edu/Macaulay2/

Singular http://www.mathematik.uni-kl.de/~zca/Singular/

Analyse numérique

Scilab http://www-rocq.inria.fr/scilab/

Matlab http://www.mathworks.com/products/matlab/mlover.shtml

Octave http://www.che.wisc.edu/octave/

CAO

VariCAD http://www.varicad.com

Système de CAO commercial pour la mécanique.

Arcad http://www.arcad.de/

Système de CAO commercial pour l'architecture.

varkon http://www.microform.se/

Système commercial de CAO haut de gamme utilisé notamment par SAB.

Alliance VHDL http://www-asim.lip6.fr/alliance/

FreeHDL http://www.freehdl.seul.org/

Green Mountain VHDL http://www.gmvhdl.com/

Systèmes libres et commerciaux de CAO pour la micro-électronique

Visualisation

Gnuplot http://www.cs.dartmouth.edu/gnuplot_info.html

fulcrum http://www.srl.caltech.edu/personnel/conrad.html

Logiciels de tracé de courbes et de visualisation scientifique.

Graphisme

Image

The Gimp http://www.gimp.org/

Programme de dessin et de manipulations d'images, similaire à *Photoshop* (mais en mieux), extensible par des ``plug-in'' et un langage de script, baptisé *Script-fu* (GPL).

Net-fu http://scheme.XCF.Berkeley.EDU/net-fu/

Création de logos *via* le Web grâce au Gimp et une applette Java.

xv http://www.trilon.com/xv/

Visualisation, de manipulation et de conversion d'images. *shareware* livré avec son code source.

xaos http://www.paru.cas.cz/~hubicka/XaoS

Pour afficher fractales et autres ensembles de Manbelbrot

3D

POV-Ray (Persistence of Vision Ray-

tracer) http://www.povray.org/

Programme de création de graphismes 3D (ray-tracer), pour Linux et les principaux OS propriétaires.

Povcenter
http://www.mime.up8.edu/~unixseb/povcentr.htm

Interface graphique pour povray 3.

Blender http://www.blender.nl/

Logiciel de design et d'animation 3D pour SGI, FreeBSD et Linux.

Moonlight Creator http://www.cybersociety.com/moonlight/

Programme de creation d'images 3D.

Video

xanim http://xanim.va.pubnix.com/

Lecture des principaux fichiers de son, d'animation ou de vidéo numérique (FLI, FLC, IFF, GIF, scripts, DL, PFX, Amiga MovieSetter, RLE, AVI, Quicktime, JFIF, MPEG, WAV, AU).

xawtv http://www.in-berlin.de/User/kraxel/linux.html

Télévision en direct sous X11.

Real Player http://www.real.com/products/player/index.html

Logiciel de lecture de séquences vidéo diffusées par Internet.

Postscript

PostScript, PDF
http://birk105.studby.uio.no/www_work/xmltools/index.html

Langage de description de page, developpé par Adobe dans les années 80 et popularisé par les imprimantes *Laser Writer* d'Apple.

GhostScript, GhostView http://www.cs.wisc.edu/~ghost/index.html

Interpréteur et visualiseur PostScript.

GV http://wwwthep.physik.uni-mainz.de/~plass/gv/

Version améliorée de GhostView.

PStill
http://www.this.net/~frank/pstill.html

Convertisseur PostScript vers Portable Document Format (PDF).

html2ps
http://www.tdb.uu.se/~jan/html2ps.html

Convertisseur HTML -> PostScript, écrit en Perl, le meilleur de son genre.

Son et musique

Linux soundapps http://www.bright.net/~dlphilp/linux_soundapps.html

Excellente liste de logiciels de musique pour Linux.

OSS/Free et OSS/Linux http://www.opensound.com/ossfree/index.html

Pilotes son pour Linux,. La version libre est intégrée normalement au noyau. La version commerciale propose des pilotes optimisés ou plus nombreux.

Traitement du son

Multitrack
http://rulhmpc38.leidenuniv.nl/private/multitrack/multitrack.html

Direct To Disk 8 pistes.

Slab
http://www.mandalux.com/soundfans/slab/index.html

Direct To Disk 64 pistes avec nombreux modules d'extension.

3MU http://www.modeemi.cs.tut.fi/~jams/3MU/

Emlateur de synthétiseur analogique TB-303.

GSyn
http://www.guildsoftware.com/gsyn/

Logiciel de synthèse sonore modulaire. Emule notamment une TB-303.

CSound
http://www.bright.net/~dlphilp/linux_csound.html

Environnement très complet de synthèse sonore par logiciel.

The Festival Speech Synthesis System
http://www.cstr.ed.ac.uk/projects/festival.html

Synthèse de la parole sous Linux.

X11Amp http://www.x11amp.bz.nu/

Lecteur de sons MP3.

MIDI

Jazz http://www.jazzware.com/

Séquenceur MIDI + audio (*shareware*).

Rosegarden http://www.bath.ac.uk/~masjpf/rose.html

Logiciel d'écriture de partitions.

Jeux et simulation

Linux Games
http://www.linuxgames.com/

Site dédié aux jeux sous Linux.

Quake 2
http://www.idsoftware.com/quake2/

Jeu d'action 3D entièrement reprogrammable.

Abuse
http://www.crack.com/games/abuse/

Jeu d'action ayant connu un très grand succès commercial. Désormais livré avec son code source ainsi que tous les outils pour développer de nouveaux jeux de plate-forme ou d'action en 3D.

xpilot http://www.xpilot.org

Jeu tactique multijoueur pour X11. Les joueurs pilotent un chasseur dans un monde artificiel et se tirent dessus avec différents types d'armes.

craft http://borneo.gmd.de/~hua/

Wargame similaire à *Dune II* et *Warcraft*.

freeciv http://freeciv.ultraviolet.org/

Clone de *Civilisation II*.

battleball
http://www.cs.utexas.edu/users/pahardin/bb.html

Jeu d'action 3-D à jouer en réseau. En gros, c'est du foot joué par des tanks.

sabre http://sabre.cobite.com/

Simulateur de vol.

nethack
http://www.win.tue.nl/games/roguelike/nethack/

Jeu de « Donjons et dragons ». Existe en version terminal et en version graphique (X11).

Emulation

Dosemu http://www.suse.com/~dosemu/

Emulateur DOS, pour système à base de processeurs Intel. Stable, utilisable pour des projets sérieux faisant appel à des applications 16 bits.

Wabi
http://www.suse.de/e/software/wabi/index.html

Emulateur MS-Windows commercial.

Citrix ICA http://download.citrix.com

Permet d'utiliser des applications Win32 gérées par un serveur Citrix.

X-MAME (MultiArcade Machine Emulator) http://drake.dit.upm.es/~ma-

me/

Emulateur de jeux d'arcades ``classique''. Permet de faire tourner plusieurs centaines de jeux, à condition de récupérer les CD-ROM, ce qui peut poser quelques problèmes légaux...

virtual-gameboy
http://www.komkon.org/fms/VGB/

Executor http://www.ardi.com/

Emulateur Macintosh commercial.

Sheepwaver
http://www.sheepshaver.com/

Emulateur Macintosh commercial pour LinuxPPC.

ibcs2 ftp://tsx-11.mit.edu/pub/linux/BETA/ibcs2/ibcs-960823.tar.gz

Permet de faire tourner des application SCO UNIX sous Linux.

vbix http://www.vbix.com/

Solutions commerciales pour porter les applications *Visual Basic* ou les scripts ASP sous Linux.

stonx
http://mips.complang.tuwien.ac.at/nino/stonx.html

Emulateur Atari ST.

Développement

Langages et compilateurs

The Free Compiler List
http://www.idiom.com/free-compilers/

FAQ des news qui recense plusieurs centaines d'implémentations gratuites (compilateurs et interpréteurs) de langages informatiques dans tous les domaines d'applications.

Review of existing Languages
http://www.tunes.org//Review/Languages.html

Liste commentée de langages de programmation, avec de très nombreux liens.

GNU http://www.gnu.org

Le temple des outils de développement libres.

egcs http://www.cygnus.com/egcs/

Compilateur multi-plate-forme hautes performances (C, C++, Objective C, FORTRAN et bientôt Java).

Objective CAML
http://pauillac.inria.fr/ocaml/

Langage à objets fortement typé. multi-plate-forme. Utilisé pour l'enseignement en France,

PGHPF
http://www.pgroup.com/ppro_hpf_desc.html

Compilateur HPF (High Performance Fortran) commercial optimisant/parallélisant pour Pentium Pro.

Squeak
http://www.create.ucsb.edu/squeak/

Environnement de programmation Small-Talk multi-plate-forme développé par Apple et proposé gratuitement au public.

SmallEiffel http://www.loria.fr/projets/SmallEiffel/

Compilateur Eiffel libre développé à l'INRIA Lorraine.

DDD http://www.cs.tu-bs.de/softech/ddd/

Superbe débogueur graphique, basé sur GDB, DBX ou XDB, pour X11.

Environnement de développement intégré

Code Crusader et Code Medic
http://www.cco.caltech.edu/~glenn/medic/

Wipeout

http://www.softwarebuero.de/wipeout-eng.html

Codewarrior
http://www.metrowerks.com

Java

Port Linux du JDK
http://www.blackdown.org/java-linux.html

Principal port du Java Developer's Kit de Sun sur les Linux Intel, PowerPC, Alpha et Sparc.

Kaffe http://www.transvirtual.com

Machine virtuelle Java portable sous GPL. Elue meilleure machine virtuelle de l'année 1998 par JavaWorld.

Japhar http://www.japhar.org/

Machine virtuelle Java portable sous GPL.

Jikes
http://www.alphaWorks.ibm.com/formula/Jikes

Compilateur Java hautes performances par IBM. *Open Source*.

TowerJ http://www.twr.com/java/java-products.html

Compilateur et environnement d'exécution hautes performances, commercial.

ClassPath http://www.classpath.org/

Projet d'environnement Java 100% libre.

NetBeans http://www.netbeans.com

Superbe environnement de développement Java avec interface graphique intégrée.

VisaJ http://www.ist.co.uk/visaj/

Environnement avancé de développement d'interface utilisateur compatible avec les spécifications JavaBeans.

Langages de script

Langages de scripts (Perl, Tcl, Python, etc.)
http://www.hex.net/~cbbrowne/languages.html

Perl http://www.perl.com/

Langage de scripts multi-plate-forme pour la manipulation de fichiers textuels.

Tcl/Tk http://www.scriptics.com/

Langage très simple à mettre en œuvre fourni avec une bibliothèque graphique très pratique et multi-plate-forme.

Python http://www.python.org/

Langage à objets extensible, très souple et multi-plate-forme.

PHP : http://www.php.net/

Langage de script multi-plate-forme pour la réalisation de pages Web (HTML) dynamiques. Permet en particulier d'accéder très facilement à des bases de données (mySQL, PostgreSQL, Oracle, Sybase, etc.).

Bibliothèques de composants

The GUI Toolkit, Framework Page
http://www.geocities.com/SiliconValley/Vista/7184/guitool.html

Liste exhaustive de librairies servant à construire des interfaces graphiques utilisateurs.

Qt http://www.troll.no/

Bibliothèque de widgettes graphiques utilisant une extension du langage C++ offrant une très grande souplesse. Une application développée avec Qt pourra être portée aisément sur tous les Unix et sur Windows. Programmable en C++ ou en Perl. De nombreux outils libres sont fournis, notamment par le projet KDE, pour faciliter le développement rapide d'interfaces graphiques et les applications multilingues.

WxWindows
http://web.ukonline.co.uk/julian.smart/wxwin/

Bibliothèque graphique C++ portable, pour X11 et la plupart des systèmes propriétaires. Une nouvelle versions (2.0) est en cours de création, qui utilisera la librairie GtK sous Unix. Une application développées avec WxWindows pourra être portée aisément sur tous les Unix et sur la plupart des systèmes propriétaires.

GTK http://www.gtk.org/

Librairies de composants graphiques extraites du logiciel de dessin et de retouches d'images Gimp. Programmable en C ainsi qu'avec de nombreux langages de script. GTK est l'un des composants de base du projet GNOME.

Bibliothèques spécialisées

Mesa
http://www.ssec.wisc.edu/~brianp/Mesa.html

Une implémentation libre de OpenGL (bibliothèque graphique 3D de Silicon Graphics) sous X11.

PVM (Parallel Virtual Machine)
http://www.epm.ornl.gov/pvm/

Le standard de fait de la programmation distribuée, aussi bien sur machines multiprocesseurs que sur réseaux de stations de travail.

MICO http://diamant-atm.vsb.cs.uni-frankfurt.de/~mico/

Broker CORBA, GPL

NAG http://www.nag.com/

Compilateurs et bibliothèques scientifiques Fortran.

Divers

The CGI Resource Index
http://www.cgi-resources.com/

Plusieurs centaines de ressources liées à CGI : scripts, documentation, jobs.

HP Eloquence
http://www.hp-eloquence.com/

Environnement de développement rapide d'applications de gestion.

Annexe E
Les partis pris

Par souci de transparence envers, voici les principaux partis pris qui ont guidé la rédaction de cet ouvrage :

1. Nous respectons la diversité des licences. La liberté des logiciels est affaire de degrés et chaque auteur est libre d'adapter cette liberté à ses objectifs lorsqu'il publie un logiciel.
2. Nous considérons que les logiciels propriétaires tournant sous Linux doivent être cités car ils renforcent la validité commerciale du système d'exploitation libre.
3. Nous citons en priorité les logiciels européens car l'Europe est très présente autour de Linux et dans le développement des logiciels libres (ex. SuSE, Qt, KDE, ssh, MySQL, etc.) ou de logiciels pour Linux (Star Office, Vshop, etc.).
4. Nous mettons en avant les distributions Linux européennes (ex. SuSE, Mandrake) car, outre le fait qu'elles sont très bien conçues et emportent les faveurs de utilisateurs en Europe et aux Etats-Unis, elles offrent aux éditeurs de logiciels européens un tremplin vers le marché mondial du logiciel.
5. Nous considérons qu'au delà des logiciels libres, l'ouverture des normes de communication et l'absence de brevets sur les logiciels sont tout aussi prioritaires, en particulier pour les logiciels propriétaires.

Annexe F
Licence Open Content

This document outlines the principles underlying the OpenContent (OC) movement and may be redistributed provided it remains unaltered. For legal purposes, this document is the license under which OpenContent is made available for use.

The original version of this document may be found at http://www.opencontent.org/opl.shtml

License

Terms and Conditions for Copying, Distributing, and Modifying Items other than copying, distributing, and modifying the Content with which this license was distributed (such as using, etc.) are outside the scope of this license.

1. You may copy and distribute exact replicas of the OpenContent (OC) as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the OC a copy of this License along with the OC. You may at your option charge a fee for the media and/or handling involved in creating a unique copy of the OC for use offline, you may at your option offer instructional support for the OC in exchange for a fee, or you may at your option offer warranty in exchange for a fee. You may not charge a fee for the OC itself. You may not charge a fee for the sole service of providing access to and/or use of the OC via a network (e.g. the Internet), whether it be via the world wide web, FTP, or any other method.

2. You may modify your copy or copies of the OpenContent or any portion of it, thus forming works based on the Content, and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified content to carry prominent notices stating that you changed it, the exact nature and content of the changes, and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the OC or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this licence, unless otherwise permitted under applicable Fair Use law.

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the OC, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the OC, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part re-

gardless of who wrote it. Exceptions are made to this requirement to release modified works free of charge under this license only in compliance with Fair Use law where applicable.

3. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to copy, distribute or modify the OC. These actions are prohibited by law if you do not accept this License. Therefore, by distributing or translating the OC, or by deriving works herefrom, you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or translating the OC.

NO WARRANTY

4. BECAUSE THE OPENCONTENT (OC) IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE OC, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE OC "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK OF USE OF THE OC IS WITH YOU. SHOULD THE OC PROVE FAULTY, INACCURATE, OR OTHERWISE UNACCEPTABLE YOU ASSUME THE COST OF ALL NECESSARY REPAIR OR CORRECTION.

5. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MIRROR AND/OR REDISTRIBUTE THE OC AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE OC, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Index

386 23, 36, 56
3D 27, 29, 47, 54, 114, 200, 202, 204, 206
486 36, 41, 45, 56, 132
68000 35, 57, 64
68030 132
8080 17

A

A/UX 64
Abuse 27
Access 109
ACL 162
Acrobat 194, 198
Acteur 102
Adabas 25, 29, 67, 215
Adaptec 66
Administration 149
Admiroutes 133
Adobe 36, 200
AFNOR 150
Afrique 132
AFS 67
AFUL 111, 126, 128, 130, 135, 145, 148
Airbus 158
AIX 64
Aladdin 81
Alcôve 99, 124, 130
Algorithme 77
Allaire 68
Allen 16, 20
Alliance VHDL 55
Allman 123
Alpha 23, 35, 57, 70, 203
ALSA 54
Altaïr 16
Amherst 126
Amiga 57
Andersen 131
Andreesen 10
Animation 202
ANSI 52
Apache 22, 30, 33, 42, 66, 77, 105, 120, 143, 211, 217
Apache Group 123
Apple 11, 18, 25, 33, 41, 61, 63, 87, 142, 173
AppleShare 25, 35, 194, 197, 214
Appletalk 25
Applications 35
Applix 26, 30
Applixware 29, 32, 37, 45
April 126, 129, 131, 135
Arcad 55
Architecture 55
Arkeia 214
ARM 23, 35, 57, 59, 65, 157, 199, 215
Artistique 80
Artplan 132
AS/400 143, 194
ASP 43, 109, 144
Association 126
Associations Professionnelles 146
AT&T 55
Atari 57
Atrid 99, 130
Audio 21, 113
Audionomérique 202
Axian 58
Azerty 34

B

Ballmer 72
Baquiast 133
BASIC 195
Bazar 119
BBS 129
Behlendorf 22
Beigbeder 128
Bénéfices 95
Bénévoles 91
Beowulf 97, 204, 207
Berkeley 50, 55, 78
Bernier 110
Bien Public 96
Bind 22, 213
Bison 52
Blender 29, 47, 200, 202
Bluetooth 127
Bogue 24, 102
Bourissou 157
Brevet 74, 77, 116, 141, 152, 168
BSA 132
BSD 12, 42, 44, 55, 78, 81, 127, 214
BSDi 56, 78
Bull 137
Bureautique 37, 41, 159, 194
Business Software Alliance 132

C

C 27, 43, 52, 56, 59, 205
C++ 27, 40, 52, 205
Caldera 21, 32, 35, 98, 126
Calibration 114
Cambridge 15
Canada 58
CAO 30, 55, 107
Cap Gemini 129
Carnot 204
Cathédrale 119
CCMail 164
CD-ROM 14, 25, 28, 30, 33, 58, 60, 98, 108, 115, 141, 159
Certification 109
CGI 42, 207, 216
Challenger 37, 44
Chen 50
Cheops 211
Chiffrement 26, 89
Chirac 131
Citoyen 119
Citrix 36, 194, 197, 200
Claranet 209
Clustering 203
Cmix 48
CNet3550
Cobalt 70, 99, 110, 116
Cobol 27
CodeCrusader 207
Code Des Marchés Publics 161
Codewarrior 68, 205, 207
Cogent 60
Cold Fusion 68
ColorSync 114
Comdex 44, 144
Commerce Électronique 216
Commodore 64 36
Communauté Urbaine De Lille 211
Communicator 10, 12, 33, 36, 45, 64, 81, 119, 194, 200
Communigate 30, 213
Compaq 57, 70, 126, 131
Computer Associates 68
Concurrence Déloyale 88
Confiance 91
Conseil 96
Consommateur 75, 153, 155
Consortium 78, 146
Constitution 1
Constructeur 69
Contrefaçon 87, 141

Convention De Berne 73
Coopératif 121, 124
Copyleft 12, 15, 18, 81
Copyright 13, 15, 78, 81, 88
Corba 41, 69, 127
Corel 37, 65, 110, 157, 161, 199, 214
Couchet 129
Cougnenc 78, 129
Coût De Migration 108
Coûts 102
Crackers 16
Cray 43, 203
Création 91
Création D'entreprise 154
CRU 212
Csound 48
Cubase 49
CULTe 131
Custom Development 52
CXR 110, 209
Cyberstudio 157
Cycle Économique 102
Cygnus 52, 59, 69, 71, 92, 97, 148, 205

D

D3 67
Danemark 89
Datafellows 209
DB2 67
DCOM 71
DDD 53, 205, 207
DDE 162
Debian 30, 35, 46, 111, 115
Déboguage 122
Débogueur 52
Décompilation 73, 77, 84, 86, 160
DECstation 57
Défauts 102
Dejanews 212
Delahaye 204
Dell 70, 99, 101, 111
Dérive 138
Développement Durable 132
Dhcp 26, 212
Diffusion 91
Digital 57, 70, 78, 126
Digital Domain119204
Digital Millenium Act 88
Directive 74
Director 49
Directory Server 65
Direct-to-disk 48
Distributeur 98
Distribution 35, 98
District Industriel 151
Divergence 123
Diversité 18
DNS 26, 46, 214
Domaine Public 77
Don 120
Donnant-donnant 91
Doom 27
DOS 36, 138, 197
DosEmu 36, 197
Drag And Drop 39
Dream 128
DRIRE 145, 162
Droit D'auteur 73, 77, 88, 165
Droit Du Logiciel 74
Dual Boot 36, 60
Dump 214
Dumping 112
Duval 33
DVD-ROM 21
Dyroff 159

E

Easter Eggs 101
EBS 60
E-business 66

Echelon 94, 210
Echo 42
Ecole Des Trois Soleils 45
Économie 91, 120
ECos 59, 71, 93, 104, 134, 144, 147
Egcs 52, 56, 92
Ego 120
Emacs 17, 52, 126, 205, 207
Emagic 165
Embarqué 148
Embrace And Extend 160
Emploi 153
Emulateur 26, 36, 200
Encyclopédie 141
Entraide 124
Erian Concept 38, 110
Eridan 199
Erlangen 159
ESC Informatique 99
Espionnage 89
Etats-Unis 74, 86, 94, 151, 158
Ethernet 21
Europe 39, 42, 59, 84, 86, 153, 193, 210
Europe-Inside 193
Executor 36
Extense 42
Externalités 149
Extreme Linux 98, 204

F

FaceMySQL 215
Fantôme 124
FAT32 21
FBI 11
Fermigier 130
Festen 34
Fiable 105, 109
FileMaker 106
Finix 131
Finlande 89, 209
Firewall 26, 29, 58, 145, 208
Flammekueche Connection 131
Fond De Brevets 87, 151
Forbes 67
Fortran 27, 53
Frame Buffer 21
FrameMaker 36, 106
Framework 53, 205
Français 87
France Telecom 42, 70, 172, 214
FreeBSD 56, 69, 71, 206
Freegate 110
Freenix 129
Free Software Foundation 12, 15, 52, 79, 126
Freeware 9, 14, 25, 47, 119, 132, 201
Front Page 56
F-Secure 210
FSF 16, 20, 22, 52, 80, 93, 125
FTP 22, 43, 212
FUD 72

G

G3 35, 64
GameBoy 36
Garde Des Sceaux 75
Gates 15, 71, 120, 131, 133
Gaulle 128
Gcc 17, 20, 52, 92, 126
Gdb 17, 52
Genquery 215
Gerber 139
Getek 42
Ghostscript 81
Gimp 29, 33, 40, 47, 71, 108, 113, 127, 144, 200, 204
Gloire 92, 95, 121
GNOME 29, 39
GNU 17, 20, 22, 52, 56, 77, 80, 92, 125, 129, 205, 207
GNU General Public Licence 12
GNU General Public License 17

GNU Plot 204
Golive 68, 157, 215
GoodNoise 95
GPL 12, 17, 40, 52, 56, 69, 78, 82, 84
Grands Comptes 149
Graphique 47
Graphiste 114
Gratification 120
Gratuit 109
Gratuité 83
Green Mountain 55
Griffin 57
Groupware 25, 42, 195, 198, 200, 214
GTK 40, 53
GUILDE 131
Guillerm 38
Gulliver 131
Gutenberg 129

H

Hackers 11, 16
Hall 126
Halloween 72, 86
Hambourg 157
Harvard 16
Haute-Savoie 46
Hébergeur 216
Helsinski 19
HFS 21
Homesteading 121
Honneur 120
Hourtin 133
HowTo 127
HP 70, 131
HP 9000 57
HP Eloquence 29
HTML 37, 43, 51, 107, 157, 160, 195, 201, 215
HTTP 22, 209
Hurd 18
Hylafax 213
Hypercard 202
Hypermédia 49

I

IBCS 32
IBM 25, 41, 44, 63, 66, 71, 81, 83, 87, 133, 139, 142
IBM Japan 71
ICA 194, 197, 200
Illinois 22
Illustration 159
IMac 33, 57
Image 54, 200
Imap 195, 212
Ind 212
Industrie 54
Infobeads 42
Infopauvre 132
Informix 29, 68, 127
Ingénierie 54
Ingres 50
Inn 22
INPI 87
Inria 130
Installation 27
Intel 17, 23, 32, 70
Intérêt Général 137
Interface 76, 88
Internet 13, 21, 25, 28, 42, 56, 59, 63, 94, 209, 212
Internet Explorer 10
Internet Net News 43, 213
Interopérabilité 74, 153, 160
Intranet 25, 44, 56, 63, 98, 101, 145, 195, 210, 212, 216
Invention 142
Investissement 115
IP 26, 212
IPSEC 26, 197, 210
IPX 25, 33
Iran 132

Irix 204
ISO 164
ITRON 59, 147

J

Japon 59, 71, 86, 141, 147, 151
Java 27, 36, 43, 52, 68, 83, 164, 194, 207
JavaBeans 208
JavaScript 195
Jazz 48, 201
JBridge 36
JDK 207
Jeu 120, 140
Jikes 68
Jini 69, 83
Jospin 131

K

Kachina 115
Kaffe 170, 200, 207
Kanji 32
KDE 29, 36, 39, 79, 123, 158, 206
KDevelop 206
Keio 95
Keso 42
Kheops 99, 110
KLyX 34, 51, 106, 204
KOffice 26, 41, 112
Konica 71
KooBase 48
Korganizer 34
Ksendfax 34
Kvoice 34
Kyocera 32

L

La Macchia 11
Lang 130, 133, 140
Latex 34, 51, 129, 205
LDAP 26, 43, 195, 212, 216
Lectra 30, 107, 140
Le Marois 44
Lernout Et Hauspie 172
Les Trois Suisses 99
LGPL 80, 82
Liaisons Louées 210
Librairie Lavoisier 214
Licence 77
Limites 108
Lintel 36
Linux 7, 18, 20, 23, 33, 37, 44, 46, 49, 52, 67, 79, 124, 127, 140, 159, 194, 205, 208, 214
Linux-Center 128
Linux-Edu 31, 46
Linux Enterprise Server 99
Linux International 21, 70, 126, 134
Linux Japan 128
Linux Journal 128
Linux Magazine 128
LinuxPPC 33, 35, 126
Linux User Group 130
Lobbying 87, 131, 133
Locabureau 101, 145, 209
Logiciels Du Soleil 110
Logiciels Propriétaires 11
L'Oréal 99
Lotus 109, 131
Lotus Notes 212
Lpr 214
Lug 130
LyX 51

M

MACH 33, 64
Macintosh 18, 32, 47, 57
MacOS 12, 24, 27, 36, 39, 42, 48, 53, 63, 71, 106, 114, 129, 138, 140, 144, 194, 198, 200, 205, 208, 213, 215
Macro-économie 94
Macsyma 54

Magic 209
MagicPoint 41
Mail 216
Mainframe 24, 194, 197, 200
Maintenance 91
Make 52
Mandala 70, 99, 130
Mandrake 30, 33, 100, 158
MandrakeSoft 100
MapInfo 162
Mapple 54
Marché 153
Marois 128
Marque 116
Marques 76
Mars 35, 214
Masquerading 26, 208
Mathematica 54
Mathématique 75, 142
Matlab 54
Maxwell 26, 34, 36, 41
McCool 22
Mesa 54, 113, 206
Messaging Server 65
Metacard 202
Metrowerks 36, 68, 205, 207
Micral 137
Micro-économie 92
Micronoyau 64
Microserveur 111
Microsoft 16, 18, 37, 41, 61, 65, 69, 71, 84, 87, 131, 139, 144, 160, 208
MIDI 48, 202
Mind Share 139, 141
Ministère De L'éducation Nationale 212
Minix 19
MIPS 23, 35
Mise À Jour 103
MIT 16, 78, 95, 126
MkLinux 33, 35, 64
MNIS 31
Modem 34
Moonlight Creator 47,113, 201
Mosaic 65
Motorola 59
Mozilla 10, 65, 81
MP3 95
MPL 81
MS Basic 17
MS-DOS 19
MTIC 145
Multimédia 108, 114, 200, 202
Multinationale 149
Multiprocesseurs 205
Multi-tâche 23
Multitrack 48, 113, 202
Multi-utilisateur 24
MuPAD 54, 204
Mutualisation 103, 115, 147
MySQL 25, 29, 49, 82, 106, 158, 166, 215

N

Nasa 97, 204
Navigator 64
NCSA 22
NDS 81
NEC 59, 71
Nestcape 18
Netatalk 213
Netbeans 54, 207
NetBSD 56, 65
Netcraft 22
Netfinity 71
NetLine 209
Netscape 10, 12, 27, 33, 36, 42, 64, 69, 81, 83, 95, 119, 130, 144, 194, 201
Netscape Public Licence 12
Netshow 71
Netware 24, 33, 98
Netwinder 65, 199, 215
Network Computer 25, 27, 65, 157, 199

News 43
Newsgroup 128, 212
NewsWatcher 92
NextStep 97
NFS 25, 38, 213
Nintendo 120
NNTP 212
Noosphère 120
Normalisation 150
Norme 77, 95, 138, 160
Notes 132
Not Invented Here 158
Nouvelles Neuves De Linux 128
Novell 24, 33, 35, 81, 98, 194, 197, 213
NPL 12, 65, 81
NSA 94
NT 24, 35, 42, 67, 72, 94, 105, 128, 144, 197, 212
NTFS 21
NTT 70, 214
Nubus 64
Numéris 209

O

O2 215
Objective C 52
Object Management Group 69, 111, 127
Office 24, 37, 45, 96, 112, 157, 162, 195
OKI 59
OLE 41
Oligopole 137, 154
OMG 69, 111, 127
Omron 32
One Stop Shop 160
OpenBSD 56, 58, 89
Open Content 193
OpenDoc 41
OpenGL 54, 206
OpenLDAP 43, 213, 217
Open Sofware Foundation4264
Open Soud System 54, 202, 206, 207
Open Source 10, 13, 41, 59, 68, 73, 77, 82, 97, 121, 127, 208
OpenStep 53
Operation Sundevil 11
Oracle 25, 29, 32, 67, 99, 101, 193
Ordinateur De Réseau 199
O'Reilly 51
Organisation 142
OS/2 25, 38, 140, 195
OSF/1 56
OSI 126
OSS 54
Otsuka Shokai 71
Ou Exclusif 141

P

Pacbase 68
Pacific High Tech 21, 31, 71, 101
Palmchip 60
Panasonic 59
PAO 23, 51
Pare-feu 79
Partenaire 101
Patches 22
PC Computing 172
PDF 198
Pentagone 16
Pentium 23, 36, 45, 58, 212, 214
Pérennité 83
Perl 22, 29, 40, 42, 50, 53, 109, 207, 212
Petit 124
Photoshop 200
PHP 29, 42, 211, 215
Pick Systems 25, 67
PingOO 31, 46
Plan 36
PlayStation 205
Plug And Play 21, 28
PME 76, 112, 147, 154, 159, 214

Politique 137
Politique Industrielle 150
Polytechnique 38
Pop 195, 212, 217
POSIX 35, 59, 130
PostgreSQL 25, 29, 49, 215
Postscript 81
Potlach 121
PovRay 47, 113
PowerMacintosh 33, 35
PowerPC 23, 32, 35, 64, 94, 203, 205
PowerPoint 18
PPX 209
Prédation 115
Prestataire D'accès 42
PrimeBase 25, 68, 215
Processeurs 35
Procmail 43
Propriétaire 85
Propriété Intellectuelle 74
Propriété Intellectuelle Chronodégradable 167
Pro Rata 149
Protocole 86
Proxy 46, 101, 209
Psion 43, 157, 205
PVM 207
Python 40, 42, 50, 53, 207, 212, 215

Q

QPL 82
Qt 30, 39, 53, 82, 166, 206
Qt Public License 82
Quake 27
Quark 112, 139
Qube 99, 214
Quel 50
Quercia 46
Quicktime 114

R

R/3 215
Raadt 58
Radiosité 47
Raid 25
Raymond 13, 72, 119, 122, 127
Ray Tracing 47
Réactivité 83, 105
Recherche 54, 148
Recycler 56
Red Hat 21, 30, 32, 35, 41, 45, 70, 97, 101, 110, 116, 126
Red Hat Package Manager 30
Régulateur 153
Rendements Croissants 137
Rentable 104
Réseau Privé Virtuel 209
Responsabilité 77, 85
Revendeur 110
Rhône-Poulenc 99
Risc 214
RNIS 21
Rosegarden 48
Routeur 25, 59, 79, 98, 209
Roxen 44, 144, 158
Royalties 17, 141
RPM 30
RTSynth 49
Russie 132
Ryobi 32

S

S/400 67
Sakamura 147
Samba 35, 213
Sane 34
SAP 215
Sarajevo 132
Scholar Net 45

Sciences 54
Scilab 54, 204
SCO 36, 55, 70, 127
SCSI 66
SCSL 81, 84, 170
SDL 207
Secret 86
Sendmail 22, 30, 43, 70, 116, 123, 211, 213, 217
Sepulchre 125
Séquenceur 202
Séquestre 87
Service 116
SGBD 67, 99, 101, 215
SGML 51
Sh 35
Shan 67
Shareware 9, 14, 48, 50, 132
SheepShaver 36
SIAG 26, 34, 36, 41
Siemens Nixdorf 70, 99
Silicon Graphics 35, 127, 204
Silicon Valley 20
Sinclair 21
SLab 29
Slackware 97
Slashdot 127
Smalltalk 27
SMB 25, 35, 194, 213
SMTP 195, 211
Software AG 67, 71
SOHO 34
Solaris 56, 195, 215
Soleau 87
Solid 25, 29, 67, 215
Solution 98
Son 54, 200
Sony 142, 205
Sound Laboratory 48, 202
Sparc 23, 35, 115
SQL 25, 207
SQL Ingénierie 99, 130
Squid 209
Ssf 210
SSH 26, 89, 194, 197, 209
Ssh2 26
SSII 99, 124, 131
SSL 29, 67
Stalker 139, 212
Stallman 12, 15,17, 22, 52, 79, 126, 129
Standard Du Marché 138
Standardisation 76
Star Division 37, 157
Star Nine 144
Star Office 26, 29, 32, 36, 41, 112, 157, 194, 198, 201
Start Up 116, 139, 158
Station De Développement 205
Station De Travail 203
Statistiques 55
ST Microélectronique 59
ST Micro-électronique 199
Stonebraker 50
Strauss-Kahn 131, 133, 138
Sun 18, 38, 53, 57, 69, 81, 87, 127, 164, 170, 207
Sun3 57
Sundevil 18
SunOS 56
Supercalculateur 23, 43, 97, 203
Supercard 49
SuSE 21, 28, 32, 35, 58, 97, 101, 105, 110, 115, 126, 158, 193, 204, 210, 214
Sybase 29, 32, 68
Sympa 43, 212, 217

T

TB303 49
Tcl 53, 109, 208
TCP/IP 26, 208
TCX 50, 96, 106
Tefal 213
Telnet 200

TeX 51, 128, 205
Textile 30, 107
The Cathedral And The Bazaar 13
Titanic 204
Tk 53
Tn3270 200
TO7 137
Torvalds 18, 40, 120, 123
Toshiba 59
Toyota 60
Transmeta 20
Transvirtual 69
Troll 41, 82
Tron 59, 93, 127
Turbo Linux 31, 35, 71
Twm 36

U

UltraSparc 35, 57
Union Européenne 96, 154, 161
Université Technologique D'Helsinki 209
Unix 12, 17, 20, 22, 24, 35, 38, 43, 58, 63, 78, 82, 97, 129, 194, 197, 206, 213
Unix Systems Laboratories 55
USB 21, 33, 57
Utilisateur 101
Utilisateur Passif 102, 105, 123
Utilisateur Actif 102, 123

V

V90 21
Valorisation 116, 148
Value Added Reseller 99
VAR 99
VA Research 69, 99
Varicad 55
Vax 57
Viabilité 102
Vidéo 113, 200
Vintage 49
VisaJ 54
VisualAge 68
Visual Basic 109, 208
Visual Tcl 208
VNC 36, 194, 198
Voila 42
VPN 210
VPS 210
VShop 217

W

W3C 95, 150, 157, 160, 164
Wall 42
Walnut Creek 58
Wanadoo 209
Wassenaar 89, 94
Web 23, 41, 46, 50, 201, 209, 211
Websphere 63, 66
Webstar 144
Windows 7, 18, 25, 27, 32, 35, 38, 43, 45, 49, 60, 82, 97, 106, 114, 129, 141, 157, 194, 197, 201, 205, 208, 213, 215
Windows CE 93, 148
WINE 34, 36
Wingz 29
Winner Take All 139, 143
Wired 19, 22
Word 37, 51, 139
Wordperfect 30, 37, 65
Workflow 26, 42, 68, 109, 215
World Wide Web Consortium 95
Write Once, Run Anywhere 53
Wu FTP 22, 43, 213, 217
WxWindows 53, 206
WYSIWYG 34

X

X 25, 27, 36, 38, 45, 56, 78, 159, 198
X86 35
XAnim 113

XCDRoast 202
Xdenu 198
Xeon 23
Xerox 173
XFree 20, 56, 200
XML 51
Xpress 113, 139

Y

Yahoo 58
Yahoo! 42
Yard 29
YaST 29
Yougoslavie 132
Young 97
Yu 50

Z

Zentek 60
Zerberus 132
Zucarelli 133
ZX81 36