

Table des matières

1	Cinq mots clés pour comprendre ce dossier	3
1.1	Système d'exploitation	3
1.2	Compilation, code source	3
1.3	Logiciel propriétaire	4
1.4	Licences	4
2	La naissance des Logiciels Libres	6
2.1	1980 : les logiciels deviennent propriétaires	6
2.2	Richard M. Stallman et la Free Software Foundation	6
2.3	Le projet GNU	8
3	Les aspects philosophiques	10
3.1	Les deux Ecoles	10
3.1.1	Opensource	10
3.1.2	Freesoftware	11
3.2	La liberté au sens kantien du terme	11
3.2.1	Free	11
3.2.2	Les quatre niveaux libertés nécessaires	11
4	De l'utopie à la reconnaissance	13
4.1	Apache, le serveur web leader du marché	13
4.2	Intérêt plus grand du marché	13
4.2.1	Pourquoi les entreprises font-elles ce choix ?	13
4.2.2	Les entreprises ou administrations qui ont franchi le pas	14
4.2.3	Et Microsoft ?	15
4.3	La perte de l'idéal	16

Introduction : êtes-vous libres ?

Dis-moi les logiciels que tu utilises, je te dirai si tu es libre ...

Vous utilisez sans doute des 'copies interdites' de logiciels¹, savez-vous les peines que vous encourez ? Savez-vous aussi que les logiciels, tels qu'ils vous sont distribués, vous interdisent de comprendre leur fonctionnement ?

Vous me répondrez sans doute que n'y aviez même pas songé et, qu'après tout, cela ne vous tracasse guère. Certaines personnes estiment pourtant qu'à cause de cela vous êtes esclaves et ont décidé de se battre passionnément pour votre liberté. En effet, les logiciels que vous utilisez, comme *MS Word*, tendent à vous aliéner en vous rendant dépendant de leur utilisation et en vous interdisant la capacité de comprendre leur fonctionnement, tel un *secret défense*.

Aujourd'hui, les *logiciels libres* vous proposent de retrouver votre liberté. Ils vous redonnent ce bien précieux car ils n'ont pas de but égoïste, vous autorisent à accroître votre culture et à faire profiter des talents de tous à la communauté. Ceci n'est plus une utopie, mais une réalité qui se démontre chaque jour davantage.

¹interdites en ce sens que vous n'avez pas payé pour obtenir la licence et donc le droit d'utilisation

Chapitre 1

Cinq mots clés pour comprendre ce dossier

1.1 Système d'exploitation

Sans lui, votre ordinateur n'est qu'un "tas de feraille". Le système d'exploitation est un groupe de programmes qui donnent vie à votre machine. Un système d'exploitation comprend des programmes de différents niveaux. Certains gèrent le matériel (lecteur de disquettes, clavier, souris, mémoire, etc.). Ce matériel est géré par des programmes bas niveau (bas en ce sens qu'ils sont plus proches du matériel) contenus, entre autres, dans le *noyau*. A cela s'ajoutent de nombreux utilitaires permettant, par exemple, la copie de fichiers ou l'édition de textes. Enfin, à un niveau encore plus haut se trouvent des programmes permettant par exemple de faire de la bureautique ou de naviguer sur l'Internet.

Il faut retenir ici qu'un système d'exploitation est composé de programmes gérant le matériel et d'utilitaires, divers et variés, et que ce tout est une base nécessaire au fonctionnement d'un ordinateur.

1.2 Compilation, code source ...

"Le code source est au code binaire ce que la recette est au gâteau"

En informatique, un programme n'est pas seulement le fruit du programmeur, celui-ci doit en fait écrire une "recette" que l'on appelle le code source. Ce code, compréhensible par l'homme, n'est pas directement exécutable, de la même façon qu'une recette est compréhensible par l'homme (ou la femme) mais n'est pas mangeable. Il faut donc passer par une phase de "cuisine", appelée compilation. En programmation, le cuisinier est remplacé par un

*Vous êtes libre de reproduire ce document sur tout support à condition Page 3
- ne pas modifier le contenu
- faire apparaître cette notice dans son intégralité, même pour une copie partielle*

logiciel, appelé compilateur, qui va fabriquer le code binaire, c'est-à-dire le programme exécutable par l'ordinateur.

Par ailleurs, de la même façon qu'un cuisinier ne dévoile pas la recette de son plat, un producteur de logiciels, comme Microsoft, n'a pas non plus pour habitude de dévoiler ses secrets et ne fournit donc que le code binaire.

C'est ici que se situe le problème majeur car on ne peut pas plus transformer un code binaire en code source qu'un gâteau en recette. Il est donc impossible d'apporter une modification au programme si on le trouve trop "fade".

1.3 Logiciel propriétaire

Un parfait exemple de logiciel propriétaire est le traitement de texte *MS Word*. Sa licence est très restrictive (copie, distribution interdite). Il est incompatible avec les logiciels de ses concurrents. Un document *Word* ne peut s'ouvrir, en principe, qu'avec *MS Word*. Même si, après de nombreux efforts, certains producteurs de traitement de texte, comme *WordPerfect* ou *StarOffice*, ont réussi à faire en sorte que leurs logiciels puissent éditer des documents ".doc", Microsoft, lui, leur met des "bâtons dans les roues" en compliquant encore plus le mode d'enregistrement à chaque nouvelle version de son logiciel. *Word* fonctionne aussi sous Mac certes, mais il ne sera sans doute jamais porté vers d'autres environnements et il est clair que Microsoft ne mettra jamais à disposition le code source ou bien les spécifications pour l'enregistrement car cela aiderait sa concurrence.

Un pareil logiciel nous rend donc *esclave*, en ce sens que le producteur nous condamne non seulement à l'utiliser comme il l'a souhaité mais surtout à n'en pas utiliser d'autre.

1.4 Licences

“ La copie des logiciels, en France, [...] est punie de deux ans d'emprisonnement et un million de Francs d'amende” Article L335-2

Peut-être ne le saviez-vous pas, mais si vous utilisez des logiciels sans en posséder la licence¹, vous risquez d'encourir ces peines ...

¹En règle générale, un logiciel est accompagné d'une licence qui définit un cadre légal d'utilisation. Par abus de langage, licence signifie aussi "preuve d'achat". Une licence se résume en une suite de règles avec lesquels il faut (en principe) être en accord pour avoir le droit d'utiliser un logiciel. Classiquement, le but de la licence est de défendre les droits de l'auteur et ce en interdisant notamment la revente de copie. Ici, il s'agit de la preuve d'achat.

Les mesures de l'article L335-2 semblent dissuasives, certes, mais aujourd'hui le nombre de personnes se trouvant en fraude est si important que cette illégalité n'apparaît pas si grave. Vous, d'ailleurs, n'avez vous jamais piraté un logiciel ?

Cela pose pourtant un problème moral et c'est en partie pourquoi de nouvelles licences ont été créées. Nous nous intéresserons, ici, particulièrement à la licence GPL (GNU General Public Licence) qui, elle, est d'un genre tout autre. En effet, celle-ci, tout en promouvant un cadre légal, permet de défendre les droits de l'auteur. Elle autorise la copie et redistribution des logiciels. Le point le plus important, en terme de liberté, est qu'elle oblige l'auteur à mettre les sources à disposition. Il devient alors possible de modifier le code source pour générer (compiler) un nouveau programme amélioré, en devant toutefois mettre à disposition le code source modifié. Les premières versions de la licence GNU GPL datent de 1985 et il a fallu attendre 1986 avant d'en avoir de la forme de celle utilisée actuellement ; elle est la plus employée dans le cadre des *logiciels libres*. Il en existe d'autres, assez proches, comme par exemple LGPL.

Chapitre 2

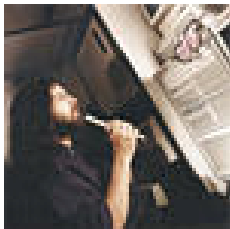
La naissance des Logiciels Libres

2.1 1980 : les logiciels deviennent propriétaires

Dans les années 70, époque où l'informatique avait plus un but scientifique que commercial, il était commun que des étudiants ou chercheurs s'échangent ou mettent à disposition des logiciels qu'ils avaient développé, en fournissant le code source. A cette époque, quand on développait un simple programme, on pouvait difficilement imaginer le vendre à une grande échelle car le marché n'était pas ce qu'il est aujourd'hui. Il existait aussi des communautés de *partage* de logiciel.

A partir des années 80, devoir signer un contrat de non diffusion devenait courant pour pouvoir utiliser un logiciel. Signer un contrat de non diffusion signifie certifier que l'on accepte de ne pas partager le logiciel et de pratiquer la rétention d'informations.

2.2 Richard M. Stallman et la Free Software Foundation



S'il y a quelqu'un d'incontournable quand on parle de *logiciels libres* c'est Richard M. Stallman (plus connu sous le nom de rms). Durant les années 70, il faisait partie d'une communauté de partage du logiciel. Considéré par tous comme le *père des logiciels libres*, il a commencé à

s'intéresser au sujet lorsqu'il travaillait au laboratoire d'intelligence artificielle au MIT¹. Le laboratoire possédait une imprimante qui tombait souvent en panne, mais comme les chercheurs avaient le code source du pilote, ils avaient pu modifier le programme afin de recevoir une notification à chaque panne. Plus tard, la société Xerox fit au laboratoire don d'une nouvelle imprimante qui, malheureusement, n'était pas plus fiable. Cette fois-ci, le code source du pilote n'étant pas fourni il devenait impossible d'ajouter un système de notification. Un peu plus tard, RMS a entendu parler d'un professeur qui possède les sources de ce pilote. S'y rendant, il lui demande s'il peut avoir une copie du code source, de façon à parer aux problèmes rencontrés avec l'imprimante. "Non, j'ai promis de ne pas vous donner de copie de ce code source." Concrètement, cela signifiait qu'il refusait de partager, comme on peut partager avec son voisin. Le laboratoire a donc naturellement été condamné à utiliser l'imprimante dans de très mauvaises conditions. Ressentant ce refus comme une agression, RMS prend alors conscience du danger de la logique propriétaire. De plus, à cette époque, la nouvelle génération d'ordinateurs de son laboratoire contient des systèmes d'exploitation pour lesquels l'utilisation implique de signer un contrat de non diffusion. Autrement, pour pouvoir utiliser ces ordinateurs il faut, à présent, promettre de ne pas partager avec son voisin. Cette accumulation de fait convainc RMS de la nécessité de réagir et ce, en développant des logiciels qui donnent aux utilisateurs la liberté.

Dans ces conditions, les choses lui apparaissent alors clairement : pour utiliser un ordinateur, il faut un système d'exploitation ; pour utiliser un système d'exploitation, il faut signer un contrat de non diffusion. Par conséquent, l'utilisation d'un ordinateur implique de perdre sa liberté. Dans ce cas, se posent deux possibilités : ne plus faire d'informatique ou devenir esclave. Heureusement, une autre solution existe : fabriquer un nouveau système d'exploitation libre. RMS, qui était développeur de système d'exploitation au MIT décide alors, fin 1983 de lancer le projet d'écrire GNU, un système d'exploitation libre. Le développement du système a commencé en Janvier 1984. Pour ne pas risquer de retrouver son travail enfermés sous des licences par le MIT, RMS décide aussi de quitter son laboratoire. Il vit donc de la vente de logiciels libres. L'argent récolté a aussi permis d'employer des développeurs. Il fonde en 1985 la FSF, Free Software Foundation, qui a pour but de récolter des fonds pour soutenir le mouvement et ce, en vendant des copies de logiciels libres et de la documentation.

¹Massachusetts Institute of Technology, Boston

2.3 Le projet GNU



Le *projet GNU* est le projet d'écrire GNU, un système d'exploitation entièrement libre, compatible avec UNIX. Bien sûr, si quiconque peut modifier un programme à souhait demeure un problème. En effet, quelqu'un pourrait très bien débbugger, améliorer et revendre un logiciel libre, profitant ainsi du travail d'autrui mais surtout, enfermant par égoïsme les libertés reçues avec ce programme. Afin de ne pas permettre cela, il à été nécessaire de créer une sorte de *copyright* revue et corrigée. Il s'agit de la licence GNU General Public Licence, GPL, citée et expliquée en introduction.

En 1990, le GNU était déjà bien avancé mais il manquait encore une pièce importante et nécessaire : le *noyau*. A vrai dire, GNU comportait un noyau, le *HURD* mais, pour des raisons techniques, le développement de celui-ci posait des problèmes.

Par chance, la communauté n'a pas eu à attendre longtemps pour avoir un noyau fonctionnel. En 1991, Linus Torvalds, a commencé à développer un noyau avec un autre design (monolithique) que celui choisi pour le noyau HURD, ce qui lui a permis d'obtenir très vite quelque chose d'exploitable. Cela avait en fait été une erreur de ne pas choisir ce design pour le noyau du GNU. Bref, dans un premier temps les développeurs du GNU n'ont pas été contactés parce que Linus Torvalds l'avait annoncé dans d'autres endroits sur internet, même s'il avait sans doute connaissance du projet GNU. Après cela, certaines personnes, ayant eu connaissance du projet se sont dit "nous avons un noyau, regardons si l'on ne pourrait pas trouver des morceaux de système d'exploitation que l'on pourrait ajouter au noyau". Ils trouvaient, bien sûr tout ce dont ils avaient besoin puisque ces outils avaient été développés pour le GNU.

Everything they needed was already available. What good fortune, they said. It's all here. Modifions ce qui est nécessaire pour assembler le tout et faire un système d'exploitation

Richard M. Stallman.

Sans même s'en rendre compte, ils assemblaient GNU et Linux, et appelaient ce système Linux ! Il est donc important de bien insister sur ce point. Aujourd'hui, on a tendance à parler de Linux pour dire GNU/Linux. Cet abus de langage est très mauvais en ce sens contribue à la confusion en faisant croire

Vous êtes libre de reproduire ce document sur tout support à condition
- *ne pas modifier le contenu*
- *faire apparaître cette notice dans son intégralité, même pour une copie partielle*

que Linux est le système entier. “Oui mais tout le monde comprend bien que quand on parle de Linux, cela ne veut plus seulement dire le noyau mais le système en entier” Je ne crois pas que dire Linux au lieu de GNU/Linux aide n’entraîne personne dans la confusion, bien au contraire. Je ne crois pas non plus que parce que “tout le monde” se trompe qu’il faut en remettre une couche. Vous savez, de très nombreuses personnes, disent que le système Windows de Microsoft est meilleur et, même s’il sont nombreux à le dire, je ne pense pas qu’il faille les croire. Enfin, je ne crois qu’il faille consulter la “sagesse populaire” dans ses choix, surtout s’il s’agit de liberté.

S’il vous plaît, ne tolérez pas cet abus de langage et expliquer pourquoi il faut dire GNU/Linux à trop nombreux utilisateurs qui n’ont pas pris le temps de se renseigner sur le système qu’ils utilisent. Si de trop nombreuses personnes commettent cette faute, on peut pas vraiment leur en vouloir car, aujourd’hui, tout contribue à faire l’amalgame. En effet, tous les magazines traitant de notre système d’exploitation ont pour nom *Linux machin* ou encore *truc Linux*. Les livres, même les plus sérieux, sont, en général, dans le faux dès la deuxième ligne de l’historique qu’ils dressent de notre système dans leurs ouvrages. En ce qui concerne les médias les choses sont encore pires. Mais le plus grave est que le but, la finalité de notre système d’exploitation, c’est à dire la *liberté* est totalement mise de côté.

Chapitre 3

Les aspects philosophiques

3.1 Les deux Ecoles

3.1.1 Opensource

Si les mouvements *Opensource* et *Freesoftware* emploient les mêmes méthodes, leurs buts sont un peu différents. Pour le premier, la priorité est la fabrication de logiciels les plus parfaits possible. La technique est simple : lorsque vous utilisez un logiciel libre, si vous rencontrez un bug, vous pouvez le notifier par *e-mail* au développeur (ou le corriger vous-même si vous en êtes capable), et très vite un correctif sera disponible sur l'internet. De la même façon, chacun peut apporter sa contribution en améliorant le programme.

Avec un produit comme *Word*, l'absence du code source rend tout cela simplement impossible. De plus la hotline de Microsoft se moquera totalement de savoir que vous avez trouvé un bug ou que vous voudriez ajouter une fonction à un logiciel. Faites le test : 0825 827 829, 0,99Fr/min

Pour le mouvement *opensource*, la priorité est donc d'avoir des logiciels de qualité irréprochable. Ce n'est pas un défi pour l'argent mais un défi intellectuel. Malheureusement, le mouvement *Opensource* s'intéresse à l'aspect technique des choses et pas à la liberté. Il est aussi en vogue, aujourd'hui, de parler de développement *Opensource* ou de logiciel *opensource*. Ce terme *Opensource* n'est pas très clair car un logiciel *Opensource*, s'il a ses sources en accès libre, n'est pas libre pour autant. S'il est accompagné d'une licence trop restrictive, l'accès aux sources, le rend tout aussi propriétaire qu'un logiciel propriétaire de base.

3.1.2 Freesoftware

Le mouvement Freesoftware n'a, bien sur, rien contre les logiciels de qualité mais a un but plus philosophe.

“Les *logiciels libres* sont une question de liberté, pas de prix” RMS

Le but est de faire des logiciels qui rendent l'utilisateur libre. Libre de copier, distribuer, étudier et modifier. Un développeur de *logiciels libres* peut être payé pour son travail. Le fruit de son travail, même s'il lui apporte un intérêt particulier (financier), apportera inévitablement aussi au bien commun. De plus, concevoir un *logiciel libre* qui est la réplique d'un logiciel propriétaire (par exemple faire un remplacement à *MS Word*) permet la désalinéation des utilisateurs et les fait ainsi tendre vers plus de liberté.

S'il s'avère que les logiciels libres sont de meilleure qualité, ce n'était pas le but recherché avant tout. Il faut garder en vue que la *liberté* reste la finalité. Par ailleurs, aussi bien dans le mouvement *freesoftware* que *opensource*, on a affaire à des programmeurs dont la passion décuple les forces.

3.2 La liberté au sens kantien du terme

3.2.1 Free

La définition du mot *free*, en anglais, est assez ambiguë et c'est sans doute une des raisons pour lesquelles beaucoup de personnes l'interprètent mal. *Free* veut aussi bien dire *gratuit* que *libre*. Pour cette raison, entre autres, nombreux sont ceux qui traduisent *freesoftware* par *logiciel gratuit*. Ceci n'est pas, à vrai dire faux, mais oublie l'essentiel. *Freesoftware* signifie, avant tout, *logiciel libre*. Même si rien ne vous empêche de donner de l'argent au développeur d'un *logiciel libre*, son caractère *libre* le rend nécessairement gratuit. Par contre, et c'est ce que la majorité des gens ne comprennent pas, un logiciel, s'il est gratuit, n'est pas libre de façon nécessaire. D'ailleurs, pour la communauté du *logiciel libre*, un logiciel gratuit et non libre est aussi mauvais qu'un logiciel propriétaire car il porte, de la même façon, atteinte à la liberté des utilisateurs. Il faut donc retenir que si la liberté implique la gratuité d'une certaine façon¹, la réciproque n'est pas toujours vraie.

3.2.2 Les quatre niveaux libertés nécessaires

Puisque la *Freesoftware Software Foundation* défend, avant tout, la liberté, elle a du définir des règles stricts auxquelles un logiciel doit obéir pour être

¹ en effet, la licence GNU GPL, n'interdit pas la vente de logiciels libres

libre. Ces quatre niveaux de liberté nécessaires et suffisants sont les suivantes :

- Liberté de copier et distribuer le programme, rendant ainsi service à autrui et par conséquent à la communauté
- Liberté d'utiliser le programme dans le cadre qu'il souhaite
- Liberté d'étudier et adapter, à volonté, le programme.
- Liberté de modifier le programme, l'améliorer et distribuer une nouvelle version, pour le bien commun.

Si un logiciel a ces quatre niveaux de liberté, alors il est libre pour vous. Si un utilisateur distribue une version binaire (directement exécutable, donc sans devoir passer par la phase de compilation) améliorée d'un logiciel libre protégé par la licence GNU GPL, il doit redonner à l'utilisateur les quatre niveaux de liberté reçus. Donc il n'est pas possible de rendre propriétaire un logiciel libre protégé. Cependant, En effet, la liberté, au sens kantien du terme, implique des règles à respecter. Cependant, certaines licences, comme la licence de X11, n'obligent pas la remise à disposition du code modifié, ce qui pose problème car dans ce cas il devient possible de rendre propriétaire un logiciel libre et c'est ce que font certaines firmes développant des systèmes d'exploitation UNIX, en ajoutant une version de X modifiée par leurs soins à leur système, sans en fournir le code source modifié. La licence GNU GPL implique donc de redonner à l'utilisateur du logiciel modifié les libertés reçues avant la modification.

“Mais pourquoi le code source d'un programme doit-il m'être accessible?”, me direz-vous. En fait, c'est la question que beaucoup se posent, même s'ils sont informaticiens. Pour être honnête, je dois avouer que cela ne risque en effet de vous servir, du moins directement. Mais, l'utilisation d'un logiciel libre n'implique pas la lecture de son code source. Ce n'est pas le *culte de l'ingénieur fou* ou encore une *invitation à lire du code source*. Non, il n'est pas nécessaire de lire du code source, mais c'est de pouvoir le faire qui importe.

Chapitre 4

De l'utopie à la reconnaissance

4.1 Apache, le serveur web leader du marché

Nombreux sont les *logiciels libres* qui ont rencontré un franc succès et sont parfois aujourd'hui des éléments nécessaires au fonctionnement de l'internet. Les exemples suivants en sont une illustration concrète. :

- Apache : Serveur web le plus utilisé au monde, plus de la moitié du marché. En deuxième position, loin derrière, IIS de Microsoft.
- Sendmail : Serveur de mail, presque le seul utilisé.
- Named : Serveur de nom, très largement leader. Named permet de résoudre un nom en adresse IP¹ et réciproquement. Ainsi, il est plus simple de retenir `www.yahoo.fr` que `194.237.109.73`. Ce logiciel est donc indispensable à l'utilisation conviviale de l'Internet.
- GNU/Linux : Sans doute le plus populaire de cette liste, ce système d'exploitation voit le nombre de ses utilisateurs augmenter de 200 à 300% chaque année et dépassera très prochainement Windows NT sur le marché des serveurs.

4.2 Intérêt plus grand du marché

4.2.1 Pourquoi les entreprises font-elles ce choix ?

Les entreprises qui utilisent ou produisent des *logiciels libres* ont en fait deux motivations majeures : la baisse des charges et l'efficacité. Pour une entreprise, le coût des licences des logiciels est énorme. Un exemple :

¹Chaque machine connectée à un réseau comme l'internet a une adresse du type 192.168.3.154. Par exemple le serveur web `fbpc2.epfl.ch` à l'adresse IP 128.178.10.31

Windows 2000 et Office 2000 coûtent à eux deux aux alentours de 3500 Francs pour un professionnel et ce n'est rien à côté de logiciels spécialisés. C'est pourquoi se tourner vers une solution libre, et donc gratuite, permet une économie non négligeable. De plus, les *logiciels libres* sont bien souvent de meilleure qualité. Ainsi, quand on a le choix entre payer des milliers de Francs par poste pour un résultat médiocre (plantages, etc) ou bien avoir une solution stable et gratuite, le choix est très vite fait et cela aussi bien pour des raisons d'efficacité que de coût.

4.2.2 Les entreprises ou administrations qui ont franchi le pas

"Si elle a reconnu l'existence de fonctions cachées dans ses produits, la société Microsoft [...] Il est vrai que, face à la difficulté d'analyser les produits dont ils ne connaissent que la partie exécutable, les chercheurs restent réservés sur de telles possibilités."
Rapport de Arthur Paecht à l'assemblée nationale

Un premier exemple amusant est celui de la Chine. Pour contrer l'impérialisme et le capitalisme américain de firmes comme Microsoft, le gouvernement chinois préfère l'utilisation des *logiciels libres*. Mais, ici, la finalité n'est pas, avant tout, la liberté. Le gouvernement français utilise déjà 600 serveurs sous GNU/Linux (pour le site web du 1er Ministre, par exemple). Certaines rumeurs disent que l'armée française souhaite aussi se tourner vers les logiciels libres car l'accès au code source permet de vérifier ce que fait vraiment le logiciel et donc de savoir s'il ne contient pas en plus de ses fonctions naturelles un "cheval de Troie", envoyant des données aux autres nations via l'Internet.. Un autre exemple est le film *Titanic*. En effet, pour la réalisation des effets spéciaux, les réalisateurs ont fait le choix de GNU/Linux pour les raisons suivantes : Windows 95 était trop instable et Windows NT trop coûteux.

Depuis quelques années, certaines entreprises ayant vu le vent tourner ont décidé de se tourner vers ce nouveau marché en proposant un service technique afin de gagner de l'argent. C'est en fait, pour elles, bien souvent, le seul moyen d'assurer leur pérennité. C'est le cas par exemple de SUN Microsystems, leader dans de nombreux domaines spécialisés en informatique. Après avoir acheté l'allemand Stardivision (développeur de StarOffice), le géant américain décide de distribuer le code source du traitement de texte sous licence GNU GPL. SUN a aussi décidé de mettre à disposition le code source de son système d'exploitation, Solaris, mais la licence accompagnant, le code est en fait trop restrictive pour que celui-ci soit libre.

Par ailleurs, d'autres géants de l'informatique comme IBM investissent beau-

coup dans les *logiciels libres*.

Cependant, il faut rester vigilant face aux licences utilisées. En effet, si des sociétés comme Netscape n'ont pas attendu "l'effet mode" pour porter leurs logiciels sous GNU/Linux, malheureusement, les licences de ces derniers en font des logiciels propriétaires.

En clair, un logiciel fonctionnant sous GNU/Linux n'est pas nécessairement libre et les exemples des logiciels propriétaires ne manquent pas. Qu'ils soient utilisés sur un système fermé ou libre, les logiciels propriétaires restent dans tous les cas gravement nuisibles à votre liberté.

4.2.3 Et Microsoft ?

"Nous porterons nos logiciels sous tout système ayant au moins deux millions d'utilisateurs" Bill Gates, début des années 1990.

Même si le système GNU/Linux a largement dépassé les deux millions d'utilisateurs, Microsoft a longtemps voulu ignorer l'existence de ce système. Il faut dire que, jusqu'à encore récemment, le géant du logiciel n'avait pas été inquiété. A présent, quand la firme de Seattle parle de GNU/Linux, c'est surtout pour essayer de le sous estimer. Pour eux, il n'est bien sur pas question de faire un portage² de logiciels et cela se comprend tout à fait car la raison la plus souvent évoquée pour ne pas "passer à Linux" est l'absence de *Word*, *Excel*, etc. Si Microsoft a su, ainsi, rendre dépendant et donc esclaves ses utilisateurs, il se pourrait bien que la le passage récent de Staroffice sous licence GPL fasse évoluer cette situation, pour les irréductibles qui ne veulent pas se passer d'outils bureautiques propriétaires.

Cependant, il existe depuis déjà longtemps des suites bureautiques sous GNU/Linux qui on permis à certains utilisateurs de trouver en eux un "ersatz" à *MS Office*. Toutefois, ces logiciels sont tout aussi propriétaires que *Word* ou *Excel* et demeurent donc aussi mauvais pour votre liberté Mais pourquoi toujours critiquer Microsoft ? En effet, il s'est créé depuis quelques temps, un *effet mode* dans le fait de critiquer Bill Gates et sa firme. Cela est idiot. Microsoft n'est qu'un exemple et il est souvent cité parce qu'il est le plus gros exemple. Netscape et son navigateur, déjà cités plus haut, sont tout aussi coupables. Toute firme développant du logiciel propriétaire est coupable au même titre que Microsoft et il n'est pas de raison de s'acharner moins sur les autres que sur le leader en la matière.

²rendre possible l'utilisation d'un logiciel sous un autre environnement

4.3 La perte de l'idéal

A première vue, si les *logiciels libres* sont un gage de qualité et que cette raison est la principale motivation pour les entreprises et les utilisateurs à se tourner vers ce nouveau modèle, il ne faut en aucun cas perdre de vue le fait que le système GNU/Linux et les *logiciels libres* ont pour but de donner la liberté. Pour l'informaticien de base le développement de *logiciels libres* n'est pas viable car il empêche toute sorte de revenu. C'est encore, ici, une mauvaise interprétation du terme *libre*. Dans *libre*, il n'est pas question de gratuité mais de liberté. "Oui mais les logiciels libres sont gratuits" me direz-vous. Oui, c'est souvent vrai, mais ce ne l'est pas nécessairement. Vous pouvez vendre du logiciel libre ou du service. Il semblerait même que certaines entreprises fassent fortune dans ce domaine.

Un exemple concret : la firme RedHat. Cette entreprise a pris de gros risques en se lançant dans la distribution du système GNU/Linux, à une époque où celui-ci n'avait pas la popularité qu'il a aujourd'hui. Elle est maintenant un des leaders sur ce marché. A première vue il apparaît donc clairement possible de gagner de l'argent avec le *logiciel libre*. Malheureusement, un point important vient à l'encontre de cette démonstration. RedHat, pour rendre le système plus accessible aux utilisateurs, a ajouté des logiciels à sa distribution. Ces logiciels, pour la plupart, ne sont pas libre. D'autres firmes pratiquent aussi cette stratégie en qualifiant ces logiciels de *valeur ajoutée*. Cet ajout de logiciels propriétaires et donc cette *trahison* donne donc une idée claire de leurs valeurs.

Les mouvements Opensource et Freesoftware (n'oubliez pas que ce sont deux mouvements bien différents) n'ont rien contre le fait de gagner de l'argent (c'est d'ailleurs ce qui a permis RMS de vivre après avoir quitté le MIT en 1984, par la vente de documentations et copies de logiciels libres à ceux qui ne pouvaient pas les télécharger sur l'internet), mais seulement si l'on ne viole pas les règles.

Mais tout ne va pas si mal. En effet, certaines distributions comme Debian GNU/Linux veulent garder le système intègre et vous permettent d'utiliser votre ordinateur avec un système d'exploitation libre. Il est aussi possible de gagner de l'argent en développant *logiciels libres* ou du service autour du *libre*. Des entreprises comme Alcove, Idealx ou VA Linux se sont spécialisées dans ce domaine et il semble jusqu'ici que l'on n'ai pas de reproches à leur faire d'un point du respect de la *liberté*.

Conclusion : la *liberté* n'est plus une utopie

Peut-être, prendrez-vous le temps de lire ma conclusion avant de vous jeter sur votre ordinateur afin de formater votre disque (éliminant ainsi toutes les chaînes qui retiennent et nuisent à votre liberté) et y installer un système libre, comme par exemple GNU/Linux. Je me réjouirai alors de vous avoir fait *sortir de la grotte*.

Peut-être êtes-vous plus modérés et pensez que si ces réflexions ne sont pas idiotes, elles ne doivent pas pour autant mener à un intégrisme.

Le problème ici est sans doute le même qu'en philosophie : il n'y a pas de réponse tranchée, pas de vérité absolue. Mais, de la même façon, si la liberté, au sens kantien du terme est sans doute impossible à atteindre, ce n'est qu'en cherchant à y tendre au maximum qu'on peut s'en approcher. C'est ainsi que la passion et l'intégrité (et non pas intégrisme) de certains a permis de réaliser un tel travail.

C'est donc pour cela qu'il faut faire ce choix de *liberté* en se séparant radicalement (ne prétendez pas vouloir agir en deux étapes car il risquerait de na pas y avoir de deuxième étape) de ses logiciels propriétaires pour ne plus utiliser que des *logiciels libres* fonctionnant sur un système *libre*.