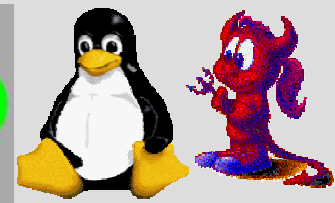


Le modèle économique du libre



On entend de plus en plus parler de «logiciels libres» ou de logiciels «OpenSource» sans généralement savoir exactement de quoi il s'agit. La confusion est à ce sujet générale: on parle de Linux comme d'un système OpenSource alors qu'il s'agit d'un système libre, on confond souvent le logiciel libre (free software) avec le logiciel gratuit (freeware) alors qu'ils sont en fait des concepts opposés. On ignore généralement que le logiciel libre existe depuis bien avant le début de l'écriture de Linux et on rate souvent le fait que les principes sous-jacents au libre ne sont pas d'ordre technique, mais bien d'ordre éthique et légaux. Les contre-vérités se multiplient, qualifiant les logiciels libres de «cancer du logiciel» ou le mouvement du libre comme «contraire aux valeurs américaine». Certains ont même voulu détruire ce mouvement en tentant de le faire interdire sous prétexte qu'il était «contraire à la constitution des U.S.A.» ou de consister en un vol de propriété intellectuelle.

Malgré tout, le modèle du libre, appliqué principalement aux logiciels, est en forte croissance: quasiment en situation de monopole dans les secteurs des applications embarquées, en Asie et dans les applications Internet, il met à mal les produits concurrents basés sur un autre modèle économique. S'étendant maintenant à d'autres secteurs, comme les documents ou les photos, le modèle économique du libre construit de nouvelles opportunités de business, certains ayant déjà abouti à de fantastiques réussites (Wikipedia, Google, RedHat, MySQL, ...)

Il est donc important pour des ingénieurs, quels que soient leurs domaines de compétences, de connaître et de comprendre le modèle du libre, afin d'en saisir les opportunités au lieu de le combattre, le plus souvent par ignorance.

La présente conférence a pour but non de défendre un modèle particulier de production et de valorisation de richesses, mais bien de l'exposer, de l'expliquer, sur base de faits et de statistiques concrètes. Les exemples sont souvent pris dans le domaine des logiciels car c'est l'origine du mouvement, mais ils sont facilement transposables à tout autre secteur d'activité ou produit où ce modèle a un sens.

Connaître n'est pas approuver. Je n'approuve pas forcément tout ce que j'expose ici.

Le modèle économique du libre

- Plan

Modèle du libre

Différences avec l'OpenSource et le freeware

Spécificités des logiciels

Exemples de modèles économiques libres

Opportunités économiques

Grandes réalisations du libre

Cette présentation débute par un rappel de ce qu'est exactement la définition du modèle du libre, tel que défini par la «Free Software Foundation». En particulier, on indiquera les différences existant entre les définitions du libre et de l'OpenSource, mouvement créé 14 ans plus tard, et le freeware (gratuitiel).

On verra ensuite ce qui fait que ce modèle est applicable aux logiciels et certains autres biens et non à l'entièreté des biens et services. On s'intéressera ensuite à des exemples de réalisations basés sur le modèle du libre, ce qui nous permettra de comprendre les mécanismes économiques qui rendent ce modèle viable.

Ceci aboutira naturellement sur des exemples d'opportunités économiques.

On terminera enfin par des exemples de grandes réalisations basées sur le modèle du libre.

Modèle du libre

- Défini par Richard Stallman (1984)

- Initialement appliqué aux logiciels
- Inspiré par le modèle économique historique des logiciels



Le modèle économique du libre a été pour la première fois formalisé par Richard Stallman en 1984 suite au refus d'un fabricant d'imprimantes de fournir le code source du driver d'un de ses modèles. Richard Stallman, chercheur au MIT, se proposait de corriger le pilote d'une imprimante achetée par l'institut, et qui produisait des messages d'erreur erronés et trompeurs en cas de pannes simples. Richard Stallman a alors débuté un combat idéologique en faveur de valeurs éthiques qui, selon lui, doivent être respectées par tous les logiciels dans l'intérêt de ses utilisateurs. (<http://www.fsf.org/philosophy/free-sw.fr.html>)

En particulier, Mr Stallman a écrit et vendu un logiciel de traitement de texte sous licence libre, Emacs. Le succès de ce mode de commercialisation d'un logiciel a été un tel succès que Mr Stallman a fondé la FSF pour en gérer les revenus et a investi ses bénéfices dans un projet d'envergure: écrire tous les logiciels nécessaires à faire tourner un ordinateur sous licence libre. Ce projet s'appelle GNU.

Modèle du libre

- Défini par Richard Stallman (1984)

- Initialement appliqué aux logiciels
- Inspiré par le modèle économique historique des logiciels



Faire mentir l'idée selon laquelle

- ✚ Les sociétés éditrices de logiciels disposent d'un droit naturel, incontestable, à être propriétaire du logiciel et à asseoir ainsi leur pouvoir sur tous ses utilisateurs, indépendamment du tort que cela cause à tous.
- ✚ Seules importent les fonctionnalités du logiciel. Les utilisateurs d'ordinateurs n'ont pas leur mot à dire quant au modèle de société qu'ils souhaitent voir mettre en place.

L'état d'esprit de Mr Stallman et ses motivations profondes résident bien dans le souci de défense de libertés et de modèle de société appliqué aux logiciels, plus qu'une volonté de mettre en place une méthode efficace de génération de logiciels de qualité. C'est une conséquence largement inattendue. Les citations ci-dessus décrivent bien les fondements de la démarche de Mr Stallman.

Modèle du libre

- Défini par Richard Stallman (1984)
- Principes fondamentaux: 4 libertés
 - La liberté d'utiliser le produit, pour tous usages
 - La liberté d'étudier le fonctionnement du produit, et de l'adapter à ses besoins
 - La liberté de redistribuer des copies
 - La liberté d'améliorer le produit, pour en faire profiter toute la communauté

Faire mentir:

« Qui partage avec son voisin est un pirate.
Qui souhaite la moindre modification
doit nous supplier de la lui faire »



Appliqué initialement au logiciel, le principe du libre selon la FSF prévoit les conditions suivantes pour pouvoir parler de logiciel libre:

- * La liberté d'exécuter le programme, pour tous les usages (liberté 0).
- * La liberté d'étudier le fonctionnement du programme, et de l'adapter à vos besoins (liberté 1).
- * La liberté de redistribuer des copies, donc d'aider votre voisin, (liberté 2).
- * La liberté d'améliorer le programme et de publier vos améliorations, pour en faire profiter toute la communauté (liberté 3).

Conséquence dans le monde du logiciel: le code source doit être rendu disponible aux utilisateurs du logiciel.

Par analogie, une voiture libre est une voiture dont on peut étudier le fonctionnement, que l'on peut utiliser comme bon nous semble, que l'on peut améliorer (tuning) et dont on peut montrer les améliorations à d'autres usagers, et dont l'usage n'interdit pas de se lancer dans la fabrication d'autres véhicules. Rien de bien abusif, on le voit. D'ailleurs, c'est bien ces libertés là qu'on a lorsque l'on achète une voiture, ou une télévision, ou tout autre produit courant. Par contre, dans le monde du logiciel, il n'est pas rare que sous des prétextes divers, le vendeur interdise l'usage dans certains cas (sur une machine multi-processeurs, sur un serveur, ou dans certains pays), interdise d'en étudier le fonctionnement ou de l'adapter (sources non communiquées, décompilation ou reverse-engineering interdits), interdise d'en distribuer des copies modifiées ou non. Ce modèle économique appliqué aux logiciels est dit «propriétaire». Il n'était pas d'application dans les débuts de l'informatique, époque où même les logiciels de base (OS, RDBMS, ...) étaient livrés sous forme de codes source, et où les contributions des usagers faisaient partie intégrante du produit final.

Les principes du libre ne sont donc pas techniques ni liés aux logiciels. Il s'agit de valeurs éthiques élémentaires respectées de tout temps pour les autres biens et donc que la FSF estime applicables aux logiciels également.

Modèle du libre

- Défini par Richard Stallman (1984)
- Principes fondamentaux: 4 libertés et quelques idéaux:
 - Développement collaboratif
 - Non privatisation des biens collectifs
 - Volonté d'imposer ce modèle par contagion
 - Protection légale par licence

Par delà les 4 conditions fondamentales, Mr Stallman et sa création, la «Free Software Foundation» (<http://www.fsf.org>), poursuivent un idéal plus vaste, couvert par le terme général de «CopyLeft» (<http://www.fsf.org/copyleft/copyleft.fr.html>).

Le premier est une conséquence naturelle de ces principes: le développement collaboratif. Lorsqu'il y a la possibilité pour chacun de participer à l'élaboration du produit final, c'est l'intérêt de l'utilisateur de le faire: il modifie ou améliore le produit pour correspondre à ses besoins. Afin d'éviter d'avoir à modifier chaque nouvelle version du produit pour y intégrer ses modifications ou améliorations, il est plus simple d'informer le concepteur initial du projet de ces modifications pour qu'il les intègre au produit standard. Ce comportement des utilisateurs est naturel et a été constaté dans le cas des logiciels dès l'apparition de développements importants, et a explosé dès l'arrivée de moyens de communications planétaires comme l'Internet, particulièrement pour des produits d'utilisation de masse. Par exemple, l'étudiant européen Linus Torvald a mis son travail d'étudiant «Minix» devenu Linux sur Internet pour avis des autres usagers, sous licence libre, et a été rapidement dépassé par les événements: la machine était lancée.

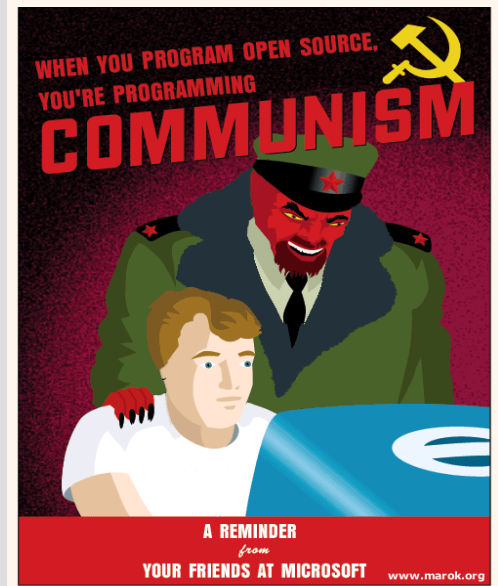
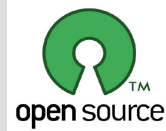
Une inquiétude majeure de la FSF est que le produit de ces développements collectifs ne soient pas confisqués dans l'intérêt égoïste d'une entreprise ou d'un particulier ne jouant pas le jeu. En particulier, il est interdit de publier tout ou partie d'un produit libre sous forme de produit propriétaire.

Plus encore, la FSF, convaincue du bien-fondé de sa démarche, poursuit une politique agressive de prosélytisme en imposant les conditions du libre à tout projet qui fasse usage d'un élément publié dans les conditions du libre. Ainsi, par contagion, la FSF aspire à la conversion de tout logiciel au modèle du libre. C'est ce qui fait dire par certains que «le logiciel libre est le cancer de l'informatique». En fait, il n'en est rien puisque, pourvu qu'on ne puise pas dans les ressources libres, on peut parfaitement fabriquer un logiciel et le distribuer sous conditions libres ou propriétaires.

Afin de garantir ces conditions et d'arriver à ces objectifs, la FSF propose une licence type, la GPL (<http://www.fsf.org/copyleft/gpl.html>, en français ici: <http://www.linux-france.org/article/these/gpl.html>) pour protéger les logiciels libres. Les logiciels libres sont donc soumis à des conditions de licence. Il ne s'agit pas du tout de logiciels «du domaine public», c'est-à-dire des logiciels publiés sans licence. Il existe aussi d'autres licences applicables à d'autres produits que les logiciels, mais poursuivant les mêmes objectifs, comme la Free Documentation License (FDL) (<http://cesarx.free.fr/gfdlf.html>) applicable aux documents écrits, y compris photos et documents sonores.

Modèle du libre

- Défini par Richard Stallman (1984)
- Principes fondamentaux: 4 libertés et quelques idéaux
- Ce que le libre n'est pas
 - Gratuit
 - Contre la propriété
 - Une doctrine politique
 - Une doctrine anti-commerciale
 - L'OpenSource (1998)



Ce que le mouvement du libre n'est pas: gratuit. Nulle-part dans les valeurs ou les définitions de la FSF il n'est imposé une quelconque restriction à la commercialisation, la vente à titre onéreux, de biens libres. Bien au contraire, les termes de la licence de prédilection de la FSF (la GPL) précise même le contraire: il est explicitement autorisé de vendre les logiciels sous cette licence, et même, il est explicitement interdit de retirer cette liberté au destinataire d'un logiciel sous cette licence. Puisqu'un freeware (gratuciel) est par définition un logiciel gratuit, il ne peut donc pas exister de logiciel qui soit à la fois libre (free software) et gratuciel (freeware). On peut parfois lire que Linux est un OS non-commercial, ou qu'un logiciel libre est gratuit. Il n'en est rien. D'ailleurs, RedHat impose des conditions d'utilisation payante pour ses distributions, pourtant complètement constituées de logiciels libres, et ce avec un succès commercial certain.

Le libre n'est pas contraire à l'idée de propriété. Le concepteur du logiciel en reste le propriétaire unique, et ne cède par la publication du logiciel sous licence GPL qu'un droit d'utilisation de celui-ci. Un bon exemple en est MySQL, qui a publié son produit phare sous licence GPL, mais n'en reste pas moins seul propriétaire du logiciel, ce qui lui permet de l'éditer et de le vendre également sous d'autres licences.

Le mouvement du libre n'est pas non plus une doctrine politique. L'objectif de la FSF n'est pas de modifier le fonctionnement de nos sociétés ou de prendre le pouvoir, mais juste de protéger les utilisateurs de logiciels contre les abus potentiels ou constatés des sociétés éditrices de logiciels propriétaires.

Les principes du libre ne sont pas non plus contraires à la commercialisation des produits. D'ailleurs, la FSF a commencé par gagner de l'argent en vendant un logiciel libre, emacs, et des sociétés développant ou distribuant du logiciel libre ou des services tournant autour du libre sont encouragées et au besoin défendues par la FSF.

Enfin, le libre, défini en 1984, n'est pas l'OpenSource, défini par l'«OpenSource Initiative» (<http://www.opensource.org/docs/definition.php>), qui est plus laxiste dans sa définition, reconnaît plus de licences comme satisfaisant ses conditions. Ils reconnaissent notamment la GPL comme licence compatible avec les valeurs de l'OpenSource ce qui lui permet de dire que «Linux est un OS OpenSource», alors que Linux est en fait un logiciel libre, publié sous licence GPL. L'OpenSource Initiative est d'ailleurs moins idéaliste, moins extrémiste, moins prosélytiste, plus proche de l'industrie du logiciel que la FSF. Les différences entre FSF et OSI sont plutôt d'ordre philosophique, mais néanmoins importantes. Les deux mouvements, au lieu de s'épauler, se détestent cordialement.

Modèle économique du libre

- Le libre, un modèle communiste ?
 - Biens communs
 - Distance vis-à-vis du modèle capitaliste
 - Vise à la défense des intérêts communs

Mais

- Pas de renoncement à la propriété
- Consentement libre
- Fonctionne même si certains ne jouent pas le jeu
- Pas applicable à tout

Le libre peut sembler correspondre aux idéaux du communisme. En effet, il s'appuie sur un effort commun pour aboutir à des biens (les logiciels libres) accessibles à tous. C'est un modèle très original, déstabilisant à première vue, pour nos esprits habitués au modèle économique basé quasi-exclusivement sur la propriété privée, le modèle capitaliste libéral classique. De plus, les motivations de la FSF ne sont pas d'ordre lucratifs, mais visent à un idéal valable pour tous. Ne s'agit-il pas là d'une forme de communisme qui ne dit pas son nom ? C'est pour cela que l'on peut lire ici où là que «le logiciel libre est contraire aux valeurs américaines» ou que le modèle du libre «appauvrit ses utilisateurs».

Rappelons que le mouvement du libre n'a rien d'une doctrine politique, ne prône pas l'abandon de la notion de propriété (respectée par la licence GPL), ne cherche pas à imposer le consentement à ses valeurs.

On peut également se demander si le modèle économique du libre est tenable, alors que celui du communisme a montré son échec à l'échelle planétaire. Là aussi, des différences entre cette doctrine politique et celle de la FSF explique les différences dans les succès rencontrés par les deux mouvements:

Le libre fonctionne même si un faible pourcentage de ses utilisateurs en respectent les règles. Le communisme, s'il peut fonctionner tout court, n'a pas pu réussir car il nécessite l'adhésion sans limite d'une majorité de la population.

Le libre ne s'applique qu'à certains biens, et non à la totalité de l'économie. Il ne cherche d'ailleurs pas à imposer un modèle économique global, juste un modèle supposé plus adéquat à la production de certains types de biens comme les logiciels et les documents.

Modèle économique du libre

- Pourquoi ce modèle pourrait-il fonctionner ?
 - Grande différence entre un logiciel et un bien matériel:

Lorsque je cède un bien matériel,
je ne l'ai plus.

Lorsque je cède un document électronique,
je l'ai encore.

Qu'est typique des logiciels, documents ou autres biens pouvant être fabriqués, commercialisés et distribués sous le modèle économique du libre ? Que différencie ces biens des autres ?

La réponse est simple mais majeure: les biens qui peuvent être cédés sans «perte» pour le cédant sont très différents au niveau économique d'un bien qui ne peut être cédé qu'aux dépens du cédant. C'est cette différence qui justifie un modèle économique original pour ces biens, et qui explique pourquoi le modèle économique du libre ne s'applique pas à tout bien.

On peut remarquer que d'autres biens ou services tombent également dans cette catégorie, et de fait sont distribués suivant des modèles économiques spécifiques, sans que ça ne choque en général. Prenons par exemple «la connaissance», qui n'est pas vendue en tant que telle mais transmise au prix de l'effort de transmission, ou d'autres biens comme le sang ou la moelle osseuse, qui ne peut être cédée que gratuitement selon la loi. Pour autant, il existe bien des écoles privées, des universités privées et des entreprises gérant les dons de sang ou d'organes et qui ne sont pas toutes des associations sans but lucratif.

Un exemple typique de conflit entre ces deux modèles économiques et entre les valeurs différentes sous-jacentes est le suivant:

Microsoft, pour intégrer les images des oeuvres majeures du musée du Louvre à son encyclopédie digitale, estimait nécessaire d'en acquérir la propriété. Ils ont donc demandé une offre pour la propriété exclusive de l'image de toutes les oeuvres de ce célèbre musée, et l'ont obtenue. Ils s'apprêtaient à sceller l'accord lorsque un responsable raisonnant suivant notre modèle européen a tiré la sonnette d'alarme, stoppant net la vente. Selon le modèle capitaliste libéral incontrôlé, tout a un prix, et Microsoft pouvait donc acheter le droit d'image de toutes les oeuvres du musée du Louvre, et ensuite les revendre au détail, dans son encyclopédie digitale ou par tout autre moyen (donne-moi ta montre, je te donnerai l'heure). Mais selon l'éthique et les valeurs humanistes européennes, les oeuvres du musée du Louvre font partie du patrimoine commun de l'humanité. Même si le bois sur lequel est peint la Joconde est bien propriété d'une institution, l'oeuvre elle-même ne saurait devenir la propriété privée d'une société étrangère. C'est exactement la même opposition qui existe entre ces deux visions de l'art et entre les modèles propriétaires et libres appliqués aux logiciels ou aux autres biens appropriés.

Modèle économique du libre

- Pourquoi ce modèle pourrait-il fonctionner ?
 - Grande différence entre un logiciel et un bien matériel:

Lorsque je cède un bien matériel,
je ne l'ai plus.

→ *besoin d'échanger le bien contre rémunération*
Lorsque je cède un document électronique,
je l'ai encore.

→ *pas obligatoirement de lien
entre cession et rémunération*

Cette particularité ouvre des perspectives intéressantes: si j'apporte une amélioration à un logiciel ou un document électronique, je peux partager cette amélioration avec les autres usagers de ce bien, sans pour autant m'appauvrir. De ce fait, si certains autres usagers font de même, tout le monde y gagne.

Un exemple en est Google. Les réponses de l'outil de recherche Google s'améliorent avec son usage. A chaque fois qu'une recherche à lieu, l'outil vérifie quel réponse l'utilisateur suit, et s'il revient sur le site de recherche ou non. Les articles les plus utilisés et qui ne provoquent pas de retour de l'utilisateur pour le même sujet sont crédités d'un poids favorable, et seront plus systématiquement proposés à l'avenir. La base de connaissance Google est donc enrichie par ses usagers. Google est également un exemple de business basé sur du libre et qui est un franc succès. En effet, Google fonctionne sur du logiciel libre et n'en est pas moins coté en bourse.

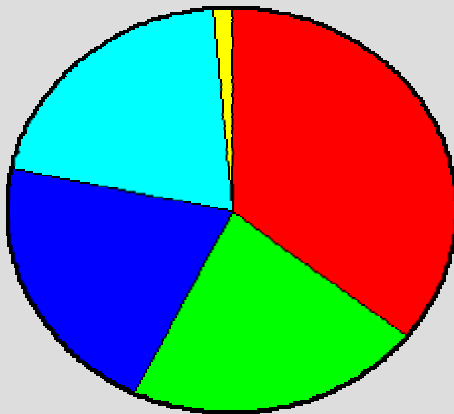
Un autre exemple en est Wikipedia, (<http://wikipedia.org>), encyclopédie libre constituée majoritairement par les contributions de ses utilisateurs. Wikipedia est maintenant une base de connaissances sans précédent, multilingue, multimédia (photos, textes, ...), distribuant des informations de grande valeur gratuitement aux usagers provenant d'autres usagers bénévoles.

Les exemples de services fonctionnant sur ce modèle son nombreux. Il existe par exemple des outils de traduction libre: les mots connus sont traduits, les autres peuvent être traduits à la main, ce qui enrichi le système. Très rapidement, un tel outil devient imbattable, y compris sur des mots de vocabulaire très spécifiques. En fait, le modèle du libre appliqué au logiciel n'est qu'un cas particulier de l'adage selon lequel l'information est plus utile lorsqu'elle est partagée.

Remarquons cependant que ce phénomène n'est vrai avec cette ampleur que pour des projets de grande envergure, impliquant énormément d'utilisateurs, car les contributeurs spontanés ne sont pas majoritaires. Ce modèle est donc moins immédiat pour des projets intéressant peu d'utilisateurs.

Modèle économique du libre

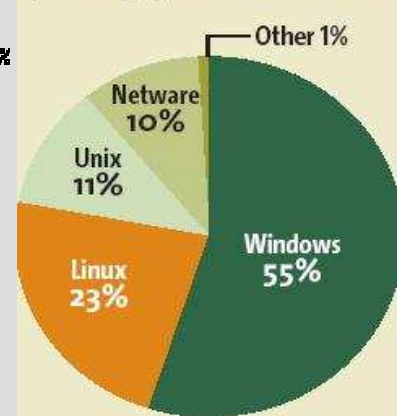
- Pourquoi ce modèle pourrait-il fonctionner ?
 - Grande différence entre un logiciel et un bien matériel
 - Ce modèle fonctionne-t-il ?



Parts de marché des OS mesurées sur Internet (mai 2000).

Source: <http://netcraft.com>

2002 Worldwide Server Operating System Market Share



SOURCE: IDC

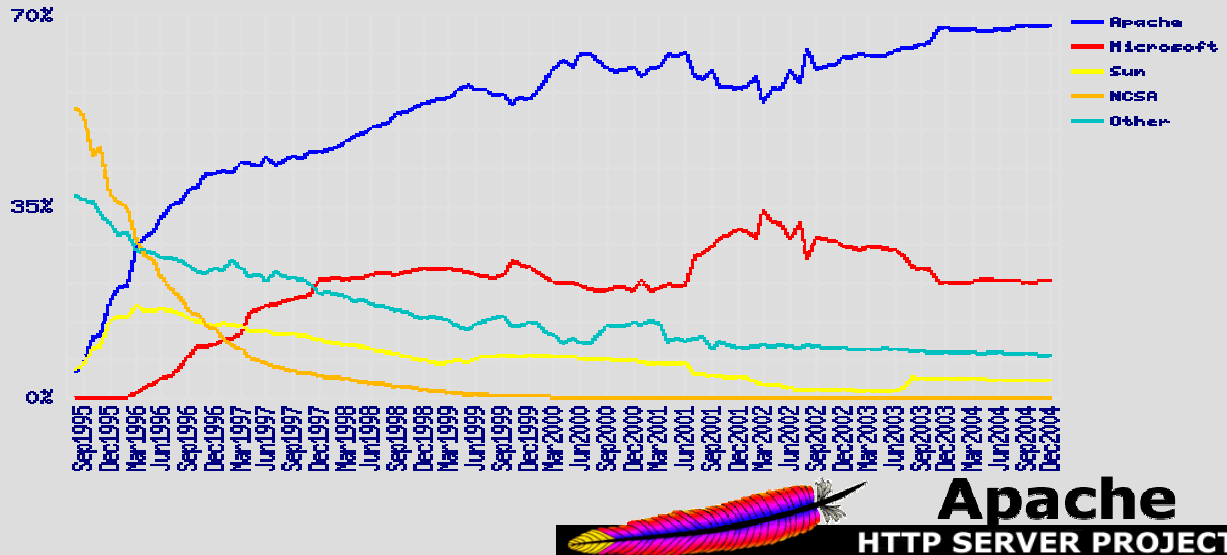
Y a-t-il des preuves que ce modèle fonctionne ?

Le développement de logiciels libres, comme Emacs ou le projet GNU, sont tellement rapides qu'ils dépassent généralement les espoirs de leurs initiateurs. Le projet GNU avait pour but de faire tourner un ordinateur entièrement sur logiciels libres. Il fallait pour cela non seulement des outils de base mais aussi un système d'exploitation. Le système d'exploitation du projet GNU a été dépassé par le développement fulgurant de Linux, pourtant un projet débuté bien après le projet GNU.

GNU/Linux, un système d'exploitation commencé en 1991 en Europe, en est un exemple de réussite: en mai 2000, il représentait plus du tiers des machines connectées à l'Internet. IDC a publié qu'en 2002, Linux à lui seul représentait 23% du marché des serveurs, en chiffre d'affaire. Vu le prix très bas des copies Linux par rapport aux autres systèmes propriétaires, on doit constater que Linux représente près d'un serveur sur deux vendu en 2002 dans le monde. 70% du marché des logiciels embarqués vendu par l'Asie en 2003 est du Linux. Il est généralement accepté que Linux est le système en plus forte croissance dans le monde, avec une augmentation de part de marché à deux chiffres par trimestre. Même sur le marché du desktop, Linux a maintenant largement dépassé le parc installé de Macintosh.

Modèle économique du libre

- Pourquoi ce modèle pourrait-il fonctionner ?
 - Grande différence entre un logiciel et un bien matériel
 - Ce modèle fonctionne-t-il ?



Mais Linux n'est pas le seul exemple de bien fabriqué sous le modèle économique du libre et qui est un franc succès: Apache en est un autre. Apache est le logiciel de serveur web le plus utilisé dans le monde. Sa part de marché est de près de 70%, laissant ses challengers loin derrière.

Ces chiffres sont à opposer avec certaines informations produites par les sociétés éditrices de logiciels propriétaires et transmises sans vérification critique par les médias selon lesquelles 95% des ordinateurs dans le monde tournent sur tel ou tel OS. Il faut bien comprendre que les biens distribués sous licence libre peuvent l'être en cascade, sans contrôle ni décompte possible de la part de l'éditeur original du produit. Il est donc impossible de compter le nombre de «licences Linux» ou de copies d'Apache, pas plus que de copies de documents libres en général. Ici, l'Internet vient à notre secours, puisqu'il est possible de mesurer à distance les parts de marché de serveurs connectés à Internet.

D'autres statistiques existent et toutes confirment la prédominance écrasante de certains produits libres sur l'Internet. Pour information, le RIPE (autorité européenne de l'Internet) a publié les parts de marché par OS mesuré sur les réseaux européens en avril 1999. Linux, à lui seul, représentait 43,3% des serveurs de fichiers connectés à l'Internet européen (source <http://www.leb.net/hzo/ioscount/data/r.9904.ftp.txt>)

Le dynamisme des développements n'est pas non plus à mettre en doute. A cette date (6/2/2005), à lui seul, le site de développement collectif <http://sourceforge.net> compte 95.060 projets en cours, réalisés par 1.007.601 utilisateurs enregistrés et Wikipedia compte déjà 461.000 articles, en ne comptant que les articles en anglais. Wikipedia existe également en 161 langues, dont le luxembourgeois et le wallon.

Modèle économique du libre

- Pourquoi ce modèle pourrait-il fonctionner ?
 - Grande différence entre un logiciel et un bien matériel
 - Ce modèle fonctionne-t-il ?
 - Avantages du modèle libre ?
 - Très grande force de travail (développement collectif par tous les consommateurs volontaires)
 - Spécification du produit et de ses améliorations par ses usagers
 - Très rapide développement et adaptation du produit aux besoins des usagers
 - Impossibilité de monopolisation d'un produit ou d'un marché
 - Pas de lien imposé entre fournisseur et usager du produit

Quels sont les avantages significatifs du modèle libre sur le modèle propriétaire, là où il s'applique ?

Tout d'abord, le travail collaboratif signifie une force de travail colossale, surtout sur des projets d'usage très général, comme un OS, une suite office, un compilateur... Aucune société ne peut en effet payer la force de travail qui se cache derrière la communauté de l'Internet, soit des millions de développeurs rémunérés chacun par leur employeur, ou travaillant à leurs heures perdues, ou encore en train de faire leurs études ou d'investir leur temps dans un projet destiné à se faire remarquer sur le marché du travail. Pour la part des travailleurs bénévoles (de moins en moins importante) il faut aussi compter sur la motivation propre: au lieu d'être obligé de se concentrer sur un projet imposé par l'employeur ou le client, le développeur bénévole se concentre sur le sujet de son choix, et organise son travail comme bon lui semble. Il en résulte une motivation que le modèle propriétaire peut difficilement obtenir.

Ensuite, l'utilisateur du produit participe à son élaboration. Il s'agit ici de l'aboutissement du principe des cercles de qualité, ou des études de satisfaction des clients. C'est pourquoi les logiciels libres, par exemple, sont souvent perçus par les utilisateurs classiques comme peu conviviaux ou ardu: ils sont développés par des techniciens, qui les trouvent parfait comme cela. Il y a donc une place pour un business qui consiste à rendre agréable et intuitif des outils par ailleurs techniquement aboutis: c'est une partie importante du rôle des distributeurs de logiciels libres.

Le cycle de développement de ces produits est très court. Quelques heures seulement après l'attentat de Madrid, le site Wikipedia proposait de nouveaux articles sur le sujet. De même, lorsqu'une faille potentielle est découverte dans un logiciel libre largement utilisé, le correctif est souvent intégré à la nouvelle version dans les heures qui suivent. Ceci est à comparer avec les délais souvent de plusieurs mois qu'il faut attendre pour obtenir des correctifs de la part d'éditeurs de logiciels propriétaires. De même, un dictionnaire libre en ligne sera très rapidement corrigé si une coquille est découverte. Ce niveau de réactivité est impossible à atteindre dans le modèle propriétaire.

Mais le plus important reste à venir: le produit libre est partagé par nature et son usage garanti donc à l'utilisateur de ne pas dépendre d'un fournisseur unique. Dans un marché comme celui du logiciel, cette propriété est très appréciable, surtout lorsqu'on suit la progression des termes des licences propriétaires, certaines relevant directement du contrat léonin ou de l'extorsion de fonds. De plus, les formats des échanges ou des données étant connus, impossible de rendre captif un usager d'un logiciel en prenant en otage ses données. Cette méthode peut paraître invraisemblable Elle est pourtant répétitivement utilisée par certaines sociétés éditrices de logiciels propriétaires pour augmenter leurs revenus sans production de valeurs en échange. On peut citer en exemple la société SCO, qui après avoir vendu du logiciel libre sous le nom «Caldera» a tenté de racketter les utilisateurs du noyau Linux sous prétexte de vol de propriété intellectuelle. Il faut savoir que le noyau Linux faisait partie de la distribution Caldera, et que SCO lui-même a donc cédé des copies des logiciels sur lesquels il revendique maintenant des droits sous licence libre...

Modèle économique du libre

- Pourquoi ce modèle pourrait-il fonctionner ?
 - Grande différence entre un logiciel et un bien matériel
 - Ce modèle fonctionne-t-il ?
 - Avantages du modèle libre ?
 - Le modèle libre n'est-il applicable qu'aux logiciels ?
 - Applicable à tous les biens ayant cette propriété
 - Appliqué depuis toujours à divers domaines:
 - Connaissances (recherche, enseignement, ...)
 - Art (musique, littérature, cinéma, ...)
 - Documents (textes, photos, sons, ...)

Le modèle économique du libre n'est pas, on l'a vu, applicable exclusivement aux logiciels. Dans l'art, par exemple, toute nouvelle oeuvre s'appuie sur le contexte de l'existant, sur les modes, les rythmes, les couleurs, les styles du moment. On peut donc considérer que tout écrivain, s'il publie un ouvrage après avoir lu au moins un autre livre, contribue à la richesse commune qu'est la culture, en ayant puisé tout d'abord quelque-chose dans le bien commun pour y ajouter sa contribution. D'ailleurs, il n'interdira pas à ses lecteurs de devenir eux-même écrivains ensuite, ce qui est très exactement une des libertés imposées par le modèle du libre: liberté d'étudier le bien, de l'améliorer et d'en redistribuer des version améliorées.

Modèle économique du libre

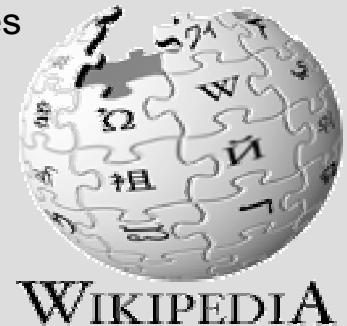
- Pourquoi ce modèle pourrait-il fonctionner ?
 - Grande différence entre un logiciel et un bien matériel
 - Ce modèle fonctionne-t-il ?
 - Avantages du modèle libre ?
 - Le modèle libre n'est-il applicable qu'aux logiciels ?

Applicable à tous les biens ayant cette propriété

Appliqué depuis toujours à divers domaines

Exemples d'applications de modèle libre:

- Wikipedia (encyclopédie libre)
- Internet
- Documentation libre



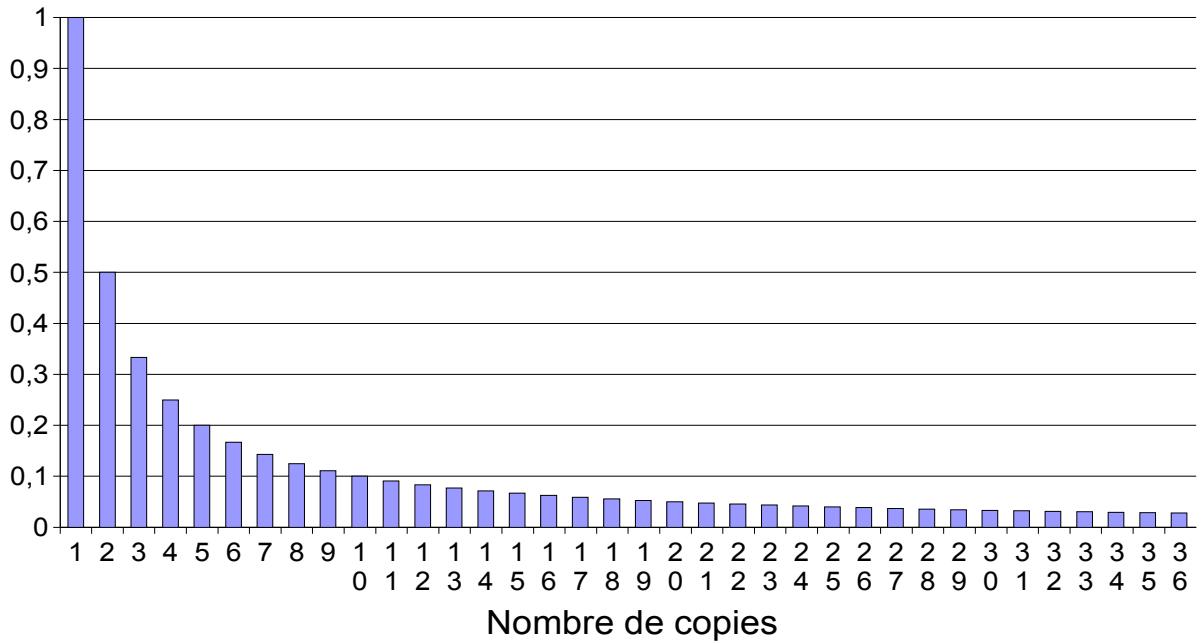
L'Internet lui-même peut être vu comme un bien qui a vu le jour suivant un modèle libre: tout d'abord commandité par un client (le département de la défense américaine), il a ensuite pu se développer du fait que les divers contributeurs (opérateurs de réseaux interconnectés partageant le même protocole IP) ont mis en commun leur infrastructure pour créer un bien commun, l'Internet, et ont vendu chacun les services découlant de ce bien commun.

Si chaque opérateur avait égoïstement facturé le trafic échangé avec les autres opérateurs, on aurait assisté à l'élaboration très lente d'un réseau parallèle au réseau téléphonique, et il est certain que l'Internet tel que nous le connaissons aujourd'hui n'existerait pas.

Ce même modèle a été suivi par Philips pour le Compact Disc. La technologie, mise au point fin des années 70, a été publiée par Philips de manière à ce que tout fabricant puisse construire et commercialiser des lecteurs compatibles avec le format que nous connaissons aujourd'hui. Philips était seul créateur du principe, et aurait pu, suivant le modèle propriétaire, créer des lecteurs Philips capables de lire des CD Philips. Sans nul doute, il aurait conservé 100% du marché du CD Philips. Les prix des lecteurs auraient été tellement élevés puisque en situation de monopole que ce marché serait resté totalement marginal, et que d'autres constructeurs (Sony, JVC, ...) auraient rapidement également mis au point leur modèle de support digital, totalement incompatible avec celui de leurs concurrents. Nous aurions donc le choix de nos jours entre 7 ou 8 types de supports, utilisable avec des lecteurs très chers, différents et incompatibles, et les maisons de disque auraient sans doute fortement hésité à investir dans une chaîne de production d'un type de support plutôt qu'un autre, tant qu'il en existait trop. Nous aurions donc sans doute encore des disques en vinyle à l'heure actuelle. Au lieu de cela, Philips a mis le standard dans le pot commun, a renoncé à un monopole, mais a certainement gagné en capturant une part de l'énorme marché créée grâce à cette sage décision.

Modèle économique du libre

Coût d'un logiciel

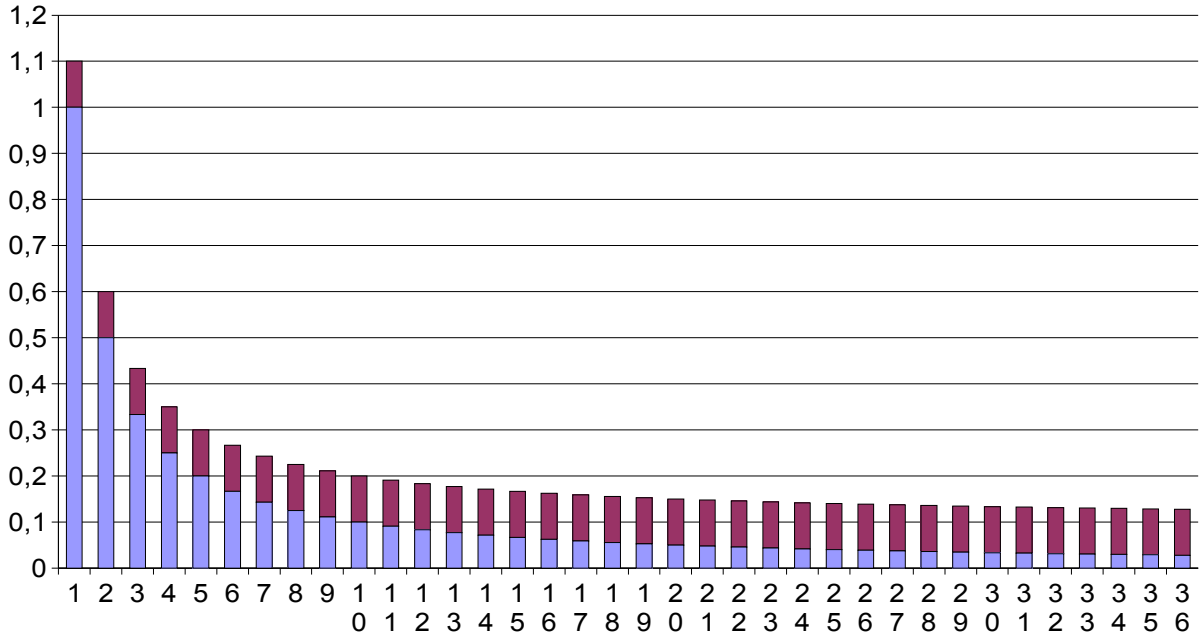


Étudions très simplement le prix de revient d'un bien produit suivant le modèle du libre, par exemple un logiciel. Le coût initial est très important: c'est de la main d'oeuvre hautement qualifiée, généralement en grande quantité. Le prix de ce bien, cependant, est divisé potentiellement par le nombre de ses consommateurs. On l'a vu, la grande différence entre un bien entrant dans le modèle du libre et un bien 'classique' réside dans le fait qu'en donnant ce bien, on le possède encore. Donc, le prix diminue suivant une fonction fortement décroissante, tendant rapidement vers 0. Tellement proche de 0 d'ailleurs, qu'on néglige en pratique ce coût en général. On confond alors classiquement le logiciel libre et le logiciel gratuit, à tort.

Mais ceci n'est qu'une image tronquée de la réalité. Si les frais de création de l'unité de départ sont effectivement divisés par le nombre d'utilisateurs, il n'en reste pas moins vrai qu'il existe les frais de copie. Ceux-ci doivent également être tenus en compte.

Modèle économique du libre

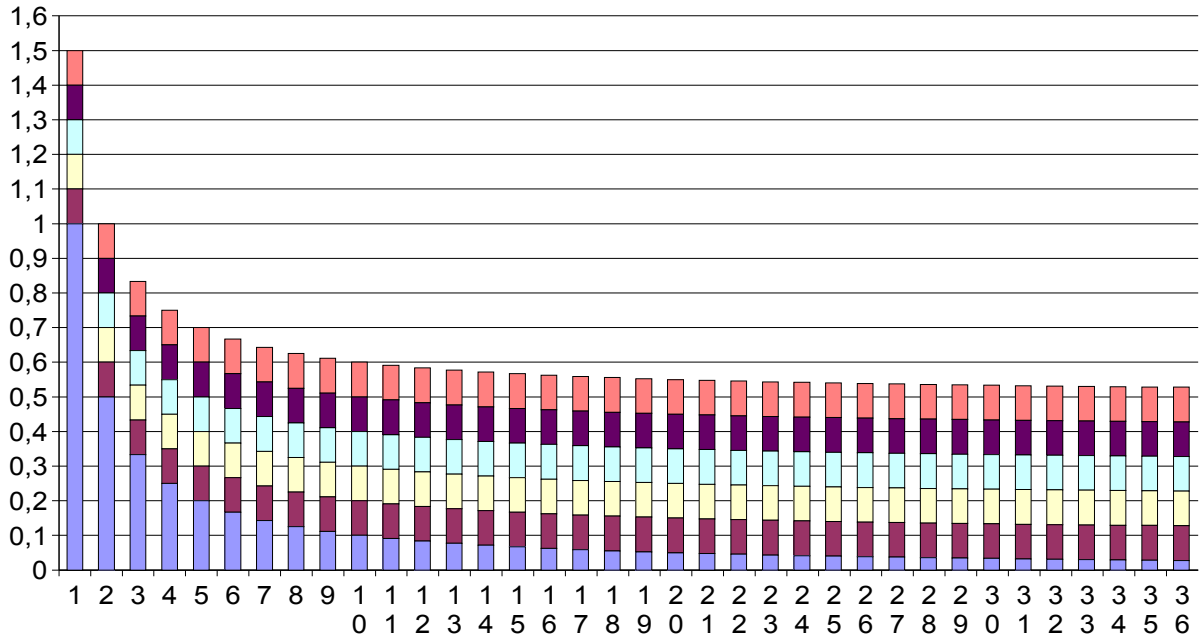
Coût d'un logiciel



Ceci montre bien que le coût d'un bien libre n'est pas en fait nul. Dans le cas de la copie d'un logiciel, cependant, ce coût est très faible et même est généralement négligé (coût d'un accès Internet). Mais il restent d'autres coûts cachés, comme

Modèle économique du libre

Coût d'un logiciel



Les formations, les coûts d'installation, de maintenance, de tuning, ... En fait, tous les coûts qui ne se divisent pas avec l'augmentation du nombre de copies deviennent très rapidement majeurs par rapport au prix de l'effort de développement. C'est pour cela que dans le cas des logiciels, le modèle économique du libre fonctionne très bien: les coûts sont généralement partagés par un très grand nombre d'utilisateurs, et les frais de copie sont négligeables. Par contre, tous les autres coûts restent, ce qui montre qu'il subsiste bien des activités lucratives dans un secteur transformé par le modèle du libre. En fait, toutes les activités qui ne sont pas mutualisables ne sont pas impactées par le modèle économique du libre.

On apprend ici que le modèle du libre ne rend pas le logiciel gratuit, mais d'un prix négligeable et généralement considéré comme nul, laissant les moyens ainsi libérés prêts à être affectés à des activités plus utiles.

Modèle économique du libre

- Le libre, une forme de dumping ?
 - Développement du logiciel est payé par ses premiers usagers (commanditaires ou initiateurs de projets)
 - Toute nouvelle occurrence d'un logiciel est vendue au prix du marché:
 - Le coût de fabrication et de distribution de l'occurrence suivante (coût de la copie)
 - Augmenté de la marge bénéficiaire du fabricant limitée par la concurrence
- ➔ Le modèle du libre est économiquement plus efficace que le modèle propriétaire pour produire certains types de biens.

Un reproche qui est souvent avancée par les sociétés souffrant de la concurrence du modèle du libre est que «vendre gratuitement est du dumping». Il faut bien ici rappeler que le libre n'implique pas du tout gratuité, que le sens du libre est un sens étique et non un sens monétaire (voir différence entre libre et gratuit). En fait, les compagnies de téléphone accusent-elles l'Internet de dumping ? Non, pourtant il est bien plus économique de téléphoner via l'Internet que via le réseau téléphonique classique. S'agit-il de dumping ? Non, puisque les sociétés qui opèrent l'Internet gagnent font des bénéfices. La réponse naturelle du marché doit donc être de garder le modèle le plus économique et de rejeter le modèle dilapidant des ressources. C'est le principe de concurrence. Dans le cas des logiciels libres, ça veut dire la faillite ou l'adaptation au nouveau modèle pour les sociétés vivant sur le modèle du logiciel propriétaire. D'ailleurs, dans le monde de la téléphonie, on constate que les plus grands opérateurs de l'Internet sont également les sociétés de téléphonie (MCI, BT, France Télécom, KPN...) et que les nouveaux opérateurs de téléphonie sont en fait des acteurs de l'Internet (Free...)

Le modèle économique du libre est plus performant pour produire certains bien que le modèle propriétaire. C'est un fait. Vouloir résister à ce fait est une voie vers la faillite. L'adaptation est la solution raisonnable.

Cela ne veut pas dire que les logiciels sont gratuits et qu'il n'y a pas moyen de gagner sa vie en faisant les logiciels. Bien au contraire. Il est possible de vendre de la maintenance, du conseil, des services, des mises-à-jour, ... A la manière de RedHat ou de Plan-Net, le logiciel libre est une formidable source de revenus qui, n'étant plus capturés par une industrie monopolistique majoritairement étrangère budgétivore, devient disponible pour du travail d'ingénieurs, de proximité, ici et maintenant, produisant des richesses adaptées aux besoins du client final.

En fait, même les sociétés dont le modèle de revenus est strictement basé sur les licences payantes d'utilisation de logiciels propriétaires n'ont qu'une très faible part de leurs employés qui travaillent directement à l'écriture du code. Toutes les autres activités (support, traduction, trainings, marketing, certifications, développements spécifiques, facilitation de l'accès aux patches, tests, matériels associés...) continuent à être exercés sous le modèle économique habituel. Seul l'effort de développement pur est affecté par le changement de modèle économique.

Modèle économique du libre

• Déroulement d'un projet

	Libre	Propriétaire
Apparition du besoin	<p>Recherche de solutions préexistantes Adaptation des solutions trouvées Participation en retour au projet commun Evolution de la solution spécifique. Développements annexes. Maintenance de l'application.</p> <p>L'utilisateur est devenu autonome, possède sa solution adaptée à ses besoins, garde son indépendance vis-à-vis d'un fournisseur unique.</p>	<p>Identification d'une opportunité de marché Décision de développer un produit propriétaire Développement par des employés non motivés sans aide d'une communauté en utilisant des outils onéreux et sans interaction avec les supposés clients Première commercialisation d'un produit conçu par une entreprise éditrice de logiciels et non par les utilisateurs. Une partie importante de l'effort est mise sur le verrouillage du futur utilisateur, sur l'apparence, sur les mesures anti-copies.</p> <p>On abouti généralement à un produit instable, obscur et peu sûr à un prix élevé</p> <p>La société cesse ses activités, ou fusionne, ou lance un autre produit incompatible, laissant les usagers du produit précédant dans les ennuis et les oblige à continuer ce cycle, car ils sont devenus captifs. Tous les efforts précédents sont perdus pour la communauté.</p>

Comparons le cycle économique d'un produit libre par rapport à un produit propriétaire. Dans le premier cas, l'identification du besoin est bien plus rapide car c'est le futur utilisateur du produit libre qui en est le concepteur, mais aussi celui qui en remarque en premier le besoin. C'est aussi celui qui a une idée la plus précise de ce qu'il a besoin (expertise de métier par exemple). Le premier bon réflexe est de regarder dans les ressources de la communauté si une solution n'existe pas déjà. Si c'est le cas (de plus en plus souvent), le problème est réglé, et à moindre frais. Si ce n'est pas le cas, il est possible généralement de trouver tout fait des parties importantes de ce qu'il va falloir construire. Celui qui a le besoin peut alors entreprendre cet effort, ou faire appel à de la main d'oeuvre externe pour le faire. Une bonne idée consiste également à tenter de trouver d'autres utilisateurs potentiels de la solution à développer, et partager l'effort entre futur utilisateurs. C'est cette démarche que l'EU recommande aux états-membres en matière de solutions informatiques pour la gestion des états.

A ce moment, une société de développement de solutions propriétaires commence seulement à identifier un marché éventuel pour un produit, et prend la décision (ou non) d'entrer dans le long processus de développement d'un produit fermé, sur base d'outils onéreux, et devant partir de rien ou devant payer un tiers pour obtenir du code qui pourrait servir. Le développement est alors réalisé non par des gens de l'art motivés, en grande majorité adhérant au modèle et travaillant de préférence sur les projets de leur choix, mais bien par des développeurs employés contraints de travailler sur un projet qu'ils n'ont pas choisi, auxquels ils n'attacheront pas leur signature. Le projet propriétaire se soucie bien moins des problèmes de sécurité (pour autant que ça ne se voie pas trop), de stabilité (pourvu que le produit fasse suffisamment illusion pour être vendu) ou de compatibilité (la société n'a pas intérêt à ce que son produit puisse être substitué facilement par un produit concurrent). De plus, une part importante de l'effort de développement consiste à protéger le produit contre la copie. Le produit propriétaire est donc beaucoup plus cher et moins approprié aux besoins de l'utilisateur final.

Ensuite, la société qui commercialise le produit propriétaire va cesser sa commercialisation pour maximiser ses revenus: le produit devient moins intéressant économiquement, et donc il faut le remplacer, même si c'est par un produit sans notable amélioration. L'utilisateur est de toutes façons captif: il est habitué au produit, se rassure du fait qu'il en connaît les limites, et a basé une part de son activité sur ce produit. Il n'a donc plus le choix. A partir de là, le prix du produit propriétaire augmente progressivement de version en version, et loin de se stabiliser ou d'évoluer vers l'intérêt de l'utilisateur, le produit évolue en fonction de l'intérêt du constructeur. A moins que celui-ci ne disparaisse par faillite ou par absorption, mettant alors en situation difficile son client et faisant perdre le bien à l'ensemble de la communauté.

Exemples de modèles libres

1. Développements initiaux

- Par idéal: Projets GNU, Gnome, Tcl/Tk, FreeDos, ...
- Par souci de protection: Linux, ...
- Par tradition: travaux universitaires, résultats de recherches...

[OpenBSD](#)



Certains projets libres, surtout au début, étaient motivés par un idéal: celui de participer au démarrage de ce modèle. Par exemple, le projet GNU de la FSF consistait strictement à l'objectif déclaré de faire tourner un ordinateur avec uniquement des logiciels libres. Financé par des dons et la vente de tels logiciels, ce projet a été un succès. C'est pour cela que le nom exact d'un système tournant sous Linux est Gnu/Linux: les outils de base livrés avec le noyau sont les outils issus du projet GNU de la FSF. Ces outils ne doivent pas être si mauvais que ça, puisque les grands constructeurs les ont repris à leur compte. On trouve ainsi de nombreux outils issus du projet GNU dans les distributions Unix de Sun, HP ou IBM.

Le projet Gnome consistait à construire un desktop libre, par réaction au desktop KDE, dépendant lui de bibliothèques non entièrement libres. C'est d'ailleurs ce desktop qui est maintenant utilisé classiquement sur les distributions Unix comme JDS ou Solaris de Sun.

De même, Wikipedia est un projet visant à construire une encyclopédie d'accès libre et gratuit, libérant le monde informatique de la main mise des encyclopédies propriétaires.

D'autres projets ont été commencés sans penser au modèle du libre, mais leur première publication a été faite sous licence GPL par soucis d'éviter de tomber dans le domaine public. Ainsi, les premières versions de Linux ont été publiées sous licence GPL, sans que Linus Torvald n'adhère totalement à l'idéal du modèle du libre.

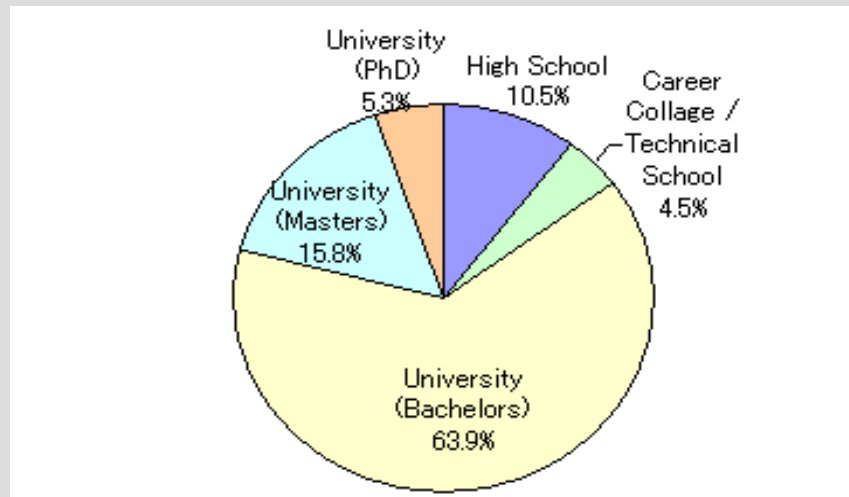
D'autres projets encore, comme les travaux de l'université de Berkeley, ont été publiés sous des licences permettant la réutilisation du code source, par tradition universitaire, sans pourtant que ces licences ne soient totalement compatibles avec les conditions édictées par la FSF. Il s'agit de la licence BSD, qui couvre le noyau BSD, à la base du nouveau Mac OSX, et qui couvre également de nombreux utilitaires intégrés dans les distributions Windows, comme le client «ftp.exe».

Dans cette catégorie, on trouve également de plus en plus de publications de recherche, sous licence libre.

Exemples de modèles libres

1. Développements initiaux

- Pas du tout des débutants:



On a généralement l'image de jeunes amateurs d'informatique débutants sans diplômes. Il n'en est rien. La composition actuelle (selon le Mitsubishi Institute en 2004) montre que les développeurs de logiciels libres se retrouvent toujours dans la communauté des universitaires, des scientifiques de haute valeur.

Exemples de modèles libres

2. Développements matures

- Obtenir le support de la communauté:

- * OpenOffice (StarOffice par Sun Microsystems)
- * Eclipse (Eclipse par IBM)
- * Mozilla (Netscape par AOL)
- * VNC (VNC par AT&T)
- * Qt (Qt de TrollTech)
- * Wikipedia, banques de photos, de documents, dictionnaires...



- Bénéficier de logiciels existants

- * Développements libres sur SourceForge ou FreshMeat
- * Logiciels fonctionnant sous KDE, MySQL, ...
- * Projets industriels (ERP, Oracle, distributions Linux, ...)



Un autre cas classique de projets qui font appel au modèle du libre, ou plus largement du modèle OpenSource, sont des projets qui, pour survivre, ont besoin de la force de travail de la communauté. On y trouve, entre autres, OpenOffice, VNC, Mozilla ou Eclipse, autant de produits initialement destinés à être des produits propriétaires, mais qui ont choisi le modèle du libre pour éviter de disparaître et pour profiter du support large de la communauté.

Un exemple intéressant est Qt, de TrollTech. Qt est une librairie tout d'abord propriétaire qui était utilisée par le desktop KDE. La communauté du libre s'est insurgée contre l'utilisation d'une librairie non libre à la base du desktop répandu sur Linux et BSD. Malgré bien des démarches, Trolltech n'a pas voulu libérer sa librairie Qt, ce qui a provoqué la création de Gnome, produit libre concurrent de KDE. Bien que commencé avec des années de retard sur KDE, Gnome a rapidement rattrapé son retard, au point que Trolltech, pour éviter que sa librairie ne soit totalement exclue du marché, l'a libéré en la publiant sous licence GPL. Depuis, KDE a repris son évolution et Trolltech vend à nouveau des outils de développement associés à sa librairie fétiche.

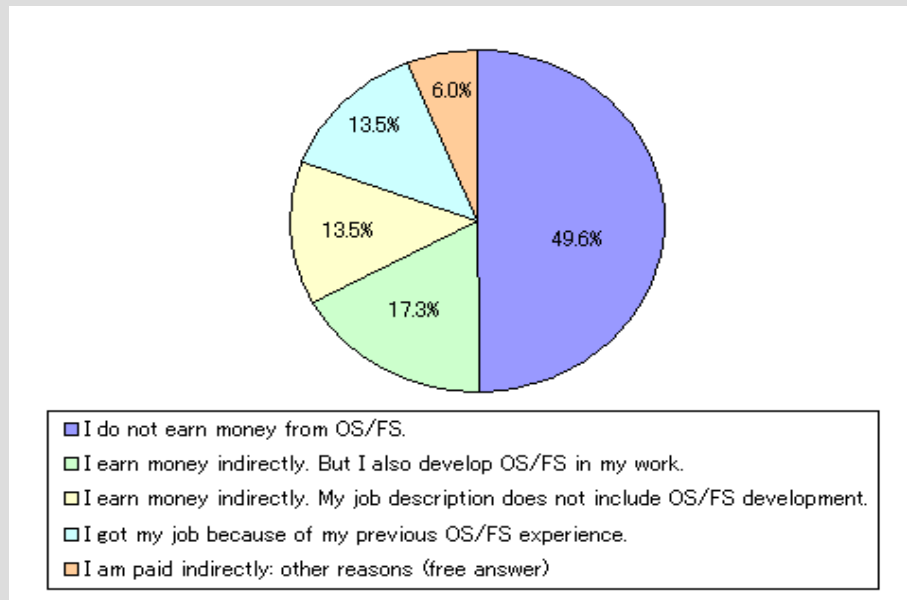
Mais le principal moteur du développement libre est de pouvoir de baser sur des projets existants. C'est en effet un immense gain de temps, et donc d'argent, de pouvoir adapter un projet existant pour correspondre à ses besoins, ou si un tel projet n'existe pas, en créer un à partir de morceaux de projets libres, et de les héberger sur un site de développement collectif comme SourceForge. On doit alors s'attendre à ce que le projet se développe tout seul, et on peut se borner à en influencer l'évolution en tant que maître d'oeuvre.

Certains projets de grande taille sont ainsi basés sur des logiciels libres, comme certains ERP, concurrents de produits propriétaires comme SAP, qui n'auraient jamais pu voir le jour sans l'apport initial du libre et le support de la communauté. D'autres produits, pourtant propriétaires, comme Oracle, sont supportés par du logiciel libre. Oracle, vers le milieu des années 90, cherchait à augmenter sa marge bénéficiaire en évitant de devoir faire payer pour son utilisation une licence de système d'exploitation. Oracle a donc commencé à développer une base de donnée autonome, tournant directement sur le matériel, sans nécessiter d'OS. Ce travail est énorme: même les fonctionnalités réduites nécessaires à une base de donnée ne sont pas facilement réalisables sur tous les types de matériels. Oracle s'est donc tourné vers le libre et maintenant officiellement recommande Linux comme plate-forme pour ses bases de données.

Exemples de modèles libres

2. Développements matures

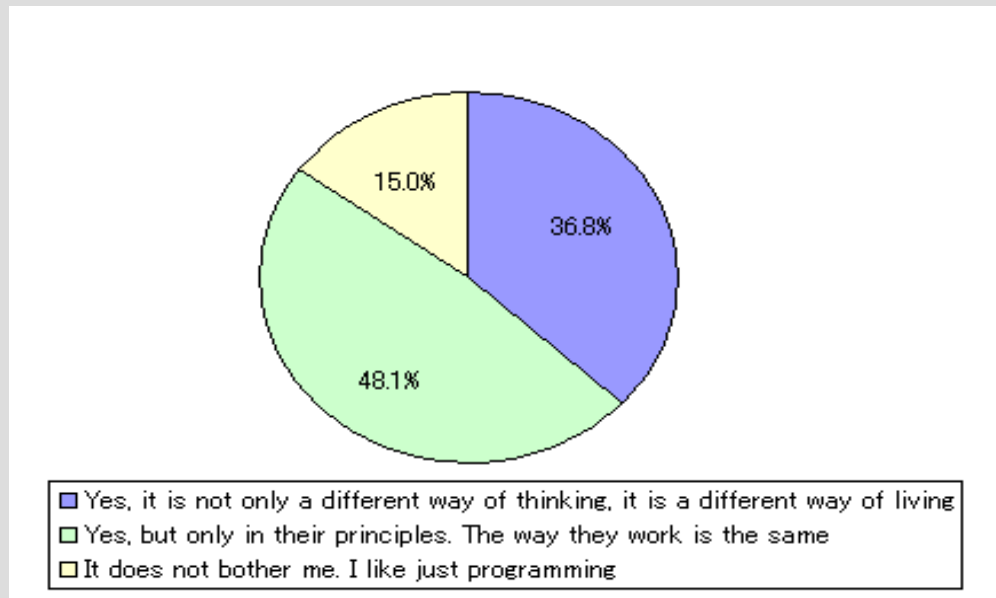
- Pas majoritairement des bénévoles



On a généralement l'image de jeunes amateurs d'informatique débutants sans diplômes. Il n'en est rien. La composition actuelle (selon le Mitsubishi Institute en 2004) montre que les développeurs de logiciels libres se retrouvent toujours dans la communauté des universitaires, des scientifiques de haute valeur.

Exemples de modèles libres

Do you feel as being part of the FS community or the OpenSource community ?



Si certaines sociétés historiquement éditrices de logiciels propriétaires lancent certains projets dans le monde libre, c'est bien entendu pour en avoir un retour aussi important que possible. Ces sociétés n'hésitent généralement pas à essayer de mettre au point une licence qui sera acceptée par l'OpenSource Initiative ou la Free Software Foundation (cette dernière étant plus difficile à convaincre) mais tout en gardant un maximum le contrôle du projet.

Cette démarche est fort risquée car, on l'a vu, des projets pourtant bien avancés comme KDE ont été sabotés par la communauté des utilisateurs sur simple base du fait qu'une partie du projet dépendait d'une librairie non libre. On voit ici la proportion de développeurs de projets libres ou OpenSource qui s'identifient comme faisant partie de l'une ou de l'autre communauté. On remarque surtout qu'une très faible proportion reste indifférente: on peut en déduire que les acteurs de ce mouvement sont très majoritairement sensibilisés aux questions éthiques qui soutiennent ces deux mouvements. Du coup, des projets lancés par des sociétés ayant des arrières-pensées non conformes à l'esprit du libre sont exposées à un grand risque que la communauté se détourne de son projet.

Exemples de modèles libres

2. Développements matures

- Garantir le contrôle de projets informatiques
- Garantir l'indépendance technologique
- Garantir la transparence des formats de données
- Améliorer l'efficacité du processus de développement

Ex: Grands projets gouvernementaux

Chine, Brésil, Allemagne, France, Espagne, Belgique, Royaume-Uni, Chili, Cuba, Union Européenne, ONU, ...

D'autres motivations non négligeables sont d'ordre stratégiques. Pour un pays, le fait de dépendre pour son infrastructure IT d'un fournisseur unique de technologie propriétaire est un problème majeur. Après que Microsoft ait été obligé de reconnaître que son OS était muni de fonctionnalités d'espionnage, imposés par l'état américain, les grands pays ont commencé à exiger un droit de regard sur le code source des programmes qu'ils utilisent. D'autres états, comme la Chine ou l'Inde, ont mis en place des projets d'envergure destinés à remplacer les programmes propriétaires par des solutions libres. Des projets de lois, et même des décrets ont été passés pour garantir la pérennité et l'accès aux données publiques au Chili. En Belgique, une loi a été proposée imposant à fonctionnalités équivalentes les solutions libres sur les solutions propriétaires. L'ONU a publié un rapport « Nationa need OpenSource », un document qui souligne que les logiciels libres et OpenSource sont nécessaires à la réduction de la fracture digitale et à l'accès des plus pauvres aux nouvelles technologies de l'information. L'Union Européenne a également publié un rapport recommandant le modèle de développement libre/OpenSource pour les applications développées par les états-membres.

Le Brésil a adopté le logiciel libre depuis longtemps, non pour des raisons idéologiques, mais simplement pour des raisons économiques. Recherchant un logiciel de travail collaboratif (un GroupWare), ils se sont vu proposés une solution propriétaire, soumise à une licence d'utilisation de 300.000 \$ pour tout le pays. Au lieu de payer cette somme à une entreprise étrangère, bénéficiant alors d'une seule version et devenant dépendant d'un fournisseur unique, le Brésil s'est penché sur ce qui était disponible dans le monde du logiciel libre. Ils ont développé leur propre « GroupWare », adapté à leurs besoins spécifiques et à leurs caractéristiques nationales (langues, coutumes, ...) pour une somme de 100.000 \$ seulement, en totalité investie dans de la main d'oeuvre locale. A partir de là, ils sont devenus autonomes, disposant des sources de leur logiciel, ce qui leur permet de le faire évoluer à leur guise. L'économie ainsi réalisée a servi à assainir des zones marécageuses et à irriguer des zones arides du pays. Le Brésil ayant bien compris le fonctionnement du libre, ils ont publié les sources de leur nouveau projet, apportant ainsi leur contribution au mouvement. Depuis, d'autres pays D'Amérique du sud ont repris le projet et l'adaptent à leur langue, leurs besoins spécifiques.

En France, Espagne, Allemagne et Angleterre, on sait que des collectivités locales ont opté pour des solutions libres par soucis d'économies. En Asie, la Chine, le Japon et la Corée du sud se sont unis pour établir un OS dédié aux besoins asiatiques: AsiaLinux, afin de limiter la dépendance de l'économie asiatique vis-à-vis des USA.

Equilibre économique du libre

- Dans le chef des utilisateurs de logiciels
 - Très forte économie en licences d'utilisation (prix à nouveau déterminés par les lois du marché)
 - Abandon de formats et plate-formes propriétaires
 - Retour de la liberté de choix entre constructeurs, éditeurs de logiciels, solutions logicielles
 - Moyens nouveaux disponible pour des développements spécifiques, des personnalisations, des améliorations, du matériel, des services
- ➔ Choix, solutions adaptées aux besoins et moyens libérés pour du matériel et des services adaptés.

Quel est l'intérêt économique du modèle libre pour les consommateurs ?

Bien entendu, une très forte réduction du prix des licences, vu le retour aux règles du marché dans la fixation des prix. Ce retour se marque également par la disparition de contraintes imposées par les conditions de licence des produits propriétaires, et par la non transparence du marché du fait de la non-interopérabilité des produits propriétaires. Ces économies sont alors disponibles pour des développements spécifiques, ce qui augmente fortement la demande en travail qualifié de proximité. En fait, les sommes non confisquées par les monopoles sont libérées pour l'artisanat local. Lorsque l'on constate que le bénéfice d'une entreprise comme Microsoft est de l'ordre de 250.000 USD par an et par employé, on comprend que ce flux monétaire qui quitte l'EU sans réelle contre-partie (les produits Microsoft ne s'améliorent pas significativement depuis dix ans ni en fiabilité ni en sécurité ni en fonctionnalités) pourrait être bien utile à payer des compétences européennes.

La question classique que se posent les consommateurs est « le libre va-t-il me coûter moins cher que le propriétaire ? ». La question n'est pas simple. Payer quelque-chose au prix du marché (ici un prix négligeable qui plus est) par rapport à payer un produit au prix imposé par le monopole laisse penser que la première solution va être la moins onéreuse. En pratique, on constate souvent qu'il n'en est rien: le budget étant de toutes façons disponible, l'économie réalisée sur le coût des licences d'utilisation va être compensé par de la consultance, du matériel plus puissant ou plus nombreux, ... Mais surtout, les fantastiques possibilités offertes par le retour du contrôle des sources dans les mains des utilisateurs va libérer l'imagination, rendre des besoins insatisfaits jusque là possibles, et créer du travail pour des développeurs de proximité, effectuant un travail utile pour le client, dédié à ses besoins, et que le modèle du libre va ensuite faire partager par la communauté.

Equilibre économique du libre

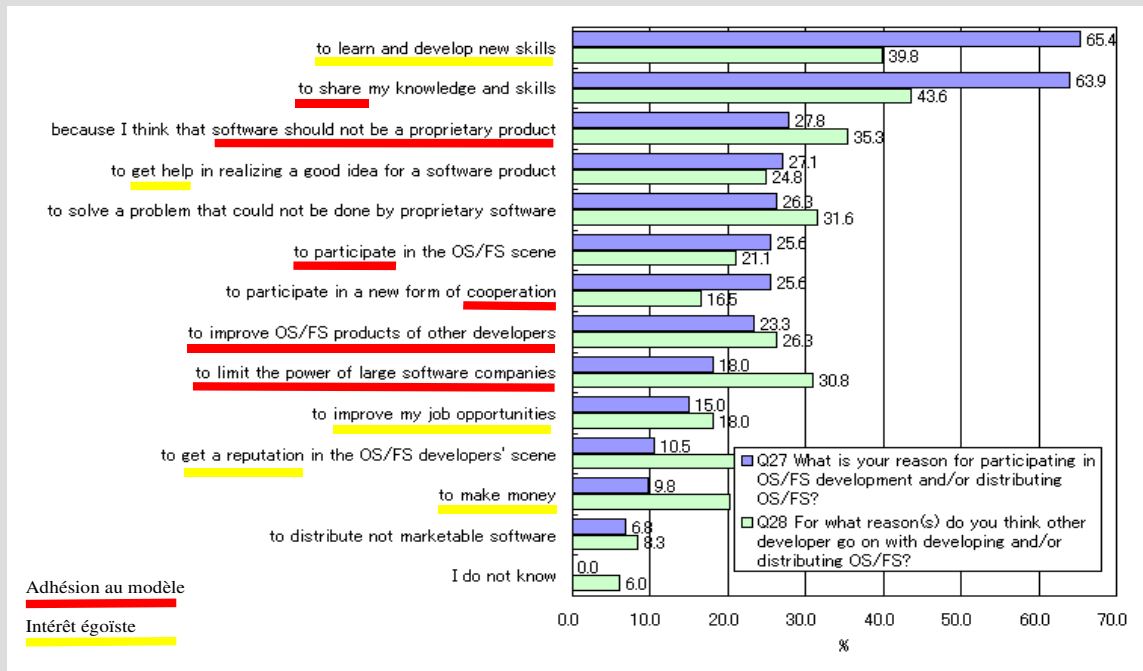
- Dans le chef des développeurs de produits libres
 - Retour de la liberté de développer sans avoir à être employé par une société monopolistique étrangère
 - Possibilité de participer et de faire aboutir des projets ambitieux
 - Grande ouverture du marché:
 - Adaptations possibles aux besoins spécifiques de clients
 - Possibilité de travailler sur des projets durables
 - ➔ Grande amélioration du marché des développements logiciels

Les développeurs trouvent dans le modèle de développement du libre une reconnaissance importante de leur travail (leurs noms font partie du projet), donc une renommée qui leur permet d'être mieux reconnus sur le marché de l'emploi. Ils peuvent également grâce aux codes sources disponibles apprendre leur métier, constater ce que font les autres, travailler sans devoir se mettre au service d'une entreprise non européenne.

Un développeur peut également trouver dans le modèle libre la possibilité de participer à des projets ambitieux, et à choisir le projet sur lequel il veut passer du temps. Il a également intérêt à ce que les sociétés utilisent des logiciels libres, puisque ça dégage des moyens pour des développements et des améliorations faites en local plutôt que d'aller enrichir le flux des capitaux envoyés à l'étranger contre de simples « titres de droit d'utilisation » que sont les licences pour logiciels propriétaires.

Equilibre économique du libre

- Motivations des développeurs de produits libres



Voici une illustration des motivations déclarées des participants au modèle de développement de logiciels libres: on constate que les motivations liées à l'adhésion au modèle sont nombreuses et parmi les plus importantes. C'est donc bien que le modèle de développement collaboratif comble ses participants, et qu'il y a donc des raisons fondamentales pour lesquelles ce système fonctionne.

Equilibre économique du libre

- Dans le chef des fabricants de produits propriétaires
 - Perte d'un marché captif
 - Fin de pratiques illégales de monopole
 - Fin de prix imposés non soumis aux lois du marché
 - ➔ Disparaître (Netscape, MacOS, Borland, Digital, ...)
 - ➔ S'adapter (Bull, Novell, IBM, ...) en modifiant son modèle économique.



Quel peut-être l'intérêt du modèle libre pour les éditeurs de produits propriétaires ? A priori aucun, comme c'est le cas des sociétés de téléphonie qui voient l'Internet manger leurs bénéfices. La seule solution, à part tenter de casser le mouvement par de la désinformation ou des procès inutiles (cas SCO): s'adapter. Adapter sa structure de revenus, en valorisant non plus le droit d'utilisation du logiciel, mais bien en vendant de la maintenance, de la formation, de la certification, ...

Prenons en exemple RedHat ou SuSE. Ces sociétés vendent essentiellement leur certification: leur distribution « entreprise » est vendue plus cher que les produits propriétaires concurrents, comme Solaris de SUN. Pourtant, leur part de marché ne cesse d'augmenter, sans doute grâce à la qualité intrinsèque d'un travail collaboratif par rapport à un travail produit dans un cercle fermé, mais aussi par l'adhésion de plus en plus répandue des gouvernements et des grandes sociétés pour un produit ouvert, inter-opérable, respectueux des standards et facilement remplaçable par un produit concurrent. SuSE et RedHat vendent également des formations, des contrats de maintenance, d'assistance téléphonique, font des développements spécifiques, ...

Dans d'autres domaines, Google vend de l'espace publicitaire et des articles sponsorisés, ainsi que son outil de recherche décliné pour les entreprises. De même, CaCert, organisme délivrant gratuitement des certificats sur Internet, vend des contrats d'assurance contre la fraude informatique, des certificats hautement sécurisés, des services d'authentification forte. Répétons-le: le modèle économique du libre n'affecte que certains aspects du monde des affaires. Il ne s'agit pas de tout casser, même si ce modèle par sa nouveauté et son efficacité est surprenant pour certains, mais il laisse la place au travail rémunéré, au bénéfices et au capitalisme dans tous les secteurs où il ne s'applique pas directement. A chacun de trouver le créneau qui s'ouvre à la suite de cette mutation, et de saisir les nouvelles opportunités ainsi créées.

Un excellent exemple de réaction appropriée de la part d'une société historiquement éditrice de logiciels propriétaires qui envisage d'évoluer pour répondre à l'émergence du modèle du libre dans son marché est illustrée ici:

« **OpenSource is the best way to build great software.** »

« The freedom that enables development in OpenSource model isn't applicable to most people deploying, so **to deploy OpenSource solutions, an expert is always required.** »

The company adds value with **packaging, documentation, support, long term insurance** that the project will continue, indemnification and more, so that the **customer can feel secure in adopting an OpenSource technology**, and feel that they are justifiably receiving a value for the money they pay. »

Opportunités économiques

- Sociétés de services basés sur des produits libres
 - conseils en matière de produits libres
 - distribution de produits libres
 - développements de produits libres
 - intégrations de produits libres
 - améliorations de produits libres
 - réglages de produits libres
 - formations sur les produits libres
 - documentation de produits libres
 - support de produits libres

Un exemple typique dans le cas du logiciel est la société de service classique SSII (Société de services en ingénierie informatique) qui se converti progressivement en SLL (Société de Services en Logiciel Libre). En abandonnant finalement pas beaucoup de revenus (les SSII ne font pas énormément de bénéfices en revendant des solutions propriétaires), elles continuent et même progressent dans toutes leurs autres activités, comme le conseil, l'aide à l'installation, les customisations, la maintenance en général.

Elles peuvent d'ailleurs fortement augmenter leur possibilités, car il y a beaucoup plus à faire sur un système informatique dont on dispose des sources que sur un système totalement fermé.

Conclusions

- Modèle économique du libre est approprié seulement à certains types de biens
- Particulièrement aux logiciels ou aux documents
- Fonctionne pour des biens largement diffusés
- Ces richesses sont le fruit d'un travail collaboratif basé sur le volontariat (non le bénévolat)
- Le modèle a prouvé son efficacité
- Il est bon pour tous ceux qui ne profitaient pas du système antérieur
- Il crée de nouvelles opportunités

Un exemple typique dans le cas du logiciel est la société de service classique SSII (Société de services en ingénierie informatique) qui se converti progressivement en SLL (Société de Services en Logiciel Libre). En abandonnant finalement pas beaucoup de revenus (les SSII ne font pas énormément de bénéfices en revendant des solutions propriétaires), elles continuent et même progressent dans toutes leurs autres activités, comme le conseil, l'aide à l'installation, les customisations, la maintenance en général.

Elles peuvent d'ailleurs fortement augmenter leur possibilités, car il y a beaucoup plus à faire sur un système informatique dont on dispose des sources que sur un système totalement fermé.

Références

La Free Software Foundation: <http://www.fsf.org>

L'OpenSource Initiative

L'encyclopédie libre: <http://fr.wikipedia.org>

Etude de Spiral <http://www.spiral.lu> sur
*Les modèles d'exploitation de l'informatique libre
au Grand-Duché de Luxembourg*

Statistiques du web: <http://www.netcraft.com>

Piège dans le Cyberspace: <http://gnuwin.epfl.ch/articles/fr/piege/>

Comparaison entre EULA et GPL:

http://members.iinet.net.au/~cybersrc/about/comparing_the_gpl_to_eula.pdf

Un site de développement collaboratif: <http://sourceforge.net>

Copyright (c) 2005 Brent Frère.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is available on the web site of the Free Software Foundation, or here: <http://www.fsf.org/licenses/fdl.html>