

# Peut-on envisager une écologie du logiciel libre favorable aux nuls ?

Nicolas Jullien, Jean-Benoît Zimmermann

▶ **To cite this version:**

Nicolas Jullien, Jean-Benoît Zimmermann. Peut-on envisager une écologie du logiciel libre favorable aux nuls ?. Terminal. Technologie de l'information, culture & société, 2006, 97-98, pp.33 - 48. hal-02136720

**HAL Id: hal-02136720**
**https://hal.science/hal-02136720**

Submitted on 22 May 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Peut-on envisager une écologie du libre favorable aux nuls ?

**Nicolas Jullien, Jean-Benoît Zimmermann***

Cela fait maintenant plusieurs années que le phénomène libre a émergé dans la sphère économique et de nombreux travaux scientifiques se sont intéressés au phénomène, parmi lesquels ceux publiés dans *Terminal* en 1999[1] ont été précurseurs. Six ans plus tard, s'il reste beaucoup à faire pour comprendre le fonctionnement des communautés de développement, certaines questions posées alors ont trouvé des éléments de réponse, notamment en ce qui concerne les motivation des développeurs individuels ou les modèles économiques.

Elles reposent toutes sur la relative homogénéité des utilisateurs et des développeurs de logiciel libre ; pour résumer, et jusqu'à récemment, le logiciel libre était affaire d'informaticiens, qui co-développaient leurs outils de travail et avaient intérêt à travailler ensemble car leurs compétences étaient complémentaires, les "clients" des entreprises du libre étant, à l'époque, plutôt les services informatiques des grands groupes [Jullien, 2003].

Ce constat est aujourd'hui remis en cause, avec la diffusion de logiciels destinés aux postes de travail, fonctionnant sous Windows, comme OpenOffice ou Mozilla et leur installation sur des ordinateurs de bureau[2]. La plupart des annonces relatives aux logiciels libres par des grands comptes porte d'ailleurs sur l'adoption de ces logiciels (villes de Munich et de Paris, Gendarmerie française…) et leur diffusion dans la population (exemple des lycées avec le Conseil régional PACA). Aujourd'hui le succès du libre finit par atteindre une catégorie de simples utilisateurs, en commençant par la part la plus dynamique d'entre eux, souvent très jeune, férue d'Internet ou proche des milieux associatifs et qui contribue à diffuser, à la fois chez les particuliers et dans les entreprises, des concepts innovants, des pratiques en évolution constante, des produits et des standards fondés sur des règles nouvelles.

Les entreprises proposant des solutions libres se tournent aussi de plus en plus vers les PME, comme en témoigne le dossier du *Journal du Net* de septembre 2005[3]. L'émergence de cette nouvelle catégorie d'utilisateurs,

* Nicolas Jullien : M@rsouin, Brest, Jean-Benoît Zimmerman : CNRS – GREQAM et IDEP, Marseille. Ce travail a été réalisé dans le cadre du projet européen "Calibre" http://www.calibre.ie

1. "Logiciels Libres : de l'utopie au marché", *Terminal*, numéro spécial N° 80-81, Automne-Hiver 1999.
2. HP vend aujourd'hui des portables sous Linux, http://svmblogs.vnunet.fr/svm/2005/09/pc_portables_so.html, disponible sur le site de Carrefour http://www.carrefourmultimedia.com/ quand, aux États-Unis, Wal-Mart vend des stations de travail http://solutions.journaldunet.com/0404/040406_walmart.shtml.
3. http://solutions.journaldunet.com/dossiers/libre/sommaire.shtml

ainsi que l'intérêt du monde marchand pour le modèle du libre, traduisent une situation nouvelle qui peut être entendue sous l'angle de l'émergence d'un marché. Mais, le corollaire est que le niveau de compétence des utilisateurs des logiciels libres baisse dans la mesure où, après une première vague de ralliement d'utilisateurs éclairés, ce sont désormais les simples utilisateurs qui rejoignent le monde du libre. Cette diffusion, voulue dès le départ par les inventeurs du concept de logiciel libre[4], nous semble remettre en cause l'écologie de l'organisation libre, fondée sur la logique du "don-contre-don"[5] (que ce soit en termes de temps passé ou de contribution financière). En moindre mal, elle ne profitera pas aux simples utilisateurs, au pire des cas, elle peut dissuader les développeurs de continuer à collaborer à ce genre de logiciels. Nous proposons dans cet article de discuter des risques de cette diffusion.

Dans les deux premières parties, nous montrerons pourquoi un nombre croissant de simples utilisateurs se rallient au logiciel libre et pourquoi la diffusion vers ces utilisateurs est effectivement une nécessité pour assurer le succès à long terme de l'organisation libre. Mais, la présence de ces simples utilisateurs ne rentre pas de manière directe dans les modalités de l'organisation actuelle du libre fondée sur la logique du don-contre-don (partie trois). Nous proposerons alors dans la partie quatre des pistes sur ce que pourrait être le nouvel équilibre de l'organisation libre.

Dans cette réflexion, nous nous appuierons sur l'étude de deux exemples de logiciels libres pour les non-informaticiens, celui d'OpenOffice et d'un ERP[6] libre, Dolibar. OpenOffice comme emblème d'un logiciel que tout le monde a ou peut avoir, avec une adoption plutôt répandue dans les grandes organisations et le soutien de grands groupes informatique, et l'ERP comme exemple de logiciel "métier" à destination des PME. Nous étaierons aussi notre analyse d'entretiens auprès de développeurs du libre que Nicolas Jullien a réalisés conjointement avec Didier Demazière et François Horn.

## Pourquoi les simples utilisateurs peuvent-ils adopter le libre ?

L'élargissement de la population concernée par les logiciels libres résulte d'un ensemble de trois facteurs complémentaires. Il s'agit, en premier lieu, de l'arrivée à un certain niveau de maturité des produits du logiciel libre, ils sont dorénavant très performants et d'un niveau de fiabilité très élevé. En deuxième lieu, cet élargissement est porté par l'extraordinaire moyen de diffusion que représente Internet et qui surajoute cette capacité aux pratiques d'interconnexion de la communauté des développeurs. En dernier lieu, la constitution d'entreprises commerciales, vouées à la distribution de logiciels libres édités avec des designs plus élaborés, des interfaces utilisateur, des manuels d'utilisation et des utilitaires d'aide, constitue un facteur décisif pour une catégorie d'utilisateurs peu technicienne et peu rompue à la navigation

4. http://www.gnu.org/gnu/manifesto.html
5. Mauss M. (1923-1924)
6. *Enterprise Resource Planning.*

sur Internet. Ce troisième type d'acteurs dans le monde du logiciel libre trouve à concilier logique commerciale et principes de non-appropriation, en fondant sa rentabilité sur la vente de services liés : "hot line", mises à jour, "débogage", ingénierie de systèmes d'information, formation...

C'est par les utilisateurs que le libre est sorti de la sphère universitaire où il a été développé. Parce que ces utilisateurs ont les mêmes besoins de garantie de la pérennité de leurs investissements, de garantie de la compatibilité des logiciels, parce qu'ils sont souvent dans un rapport de dépendance vis-à-vis des producteurs de logiciels en situation dominante, ces utilisateurs peuvent être intéressés par l'adoption du libre.

Pour les utilisateurs dotés de capacités informatiques avancées, la disponibilité des codes sources leur permet de pourvoir directement à l'adaptation des logiciels libres à leurs besoins ou situations spécifiques. Ces adaptations peuvent aussi le cas échéant comprendre le développement de nouvelles fonctionnalités. Dans tous ces cas, ces utilisateurs "sophistiqués" deviennent alors eux-mêmes développeurs. Ainsi que le montrent Lakhani et von Hippel (2000), dans de nombreuses situations, il peut être plus intéressant pour eux de contribuer aux efforts collectifs de développement du libre en publiant les modifications qu'ils y apportent, plutôt que de les garder secrètes.

Pour les simples utilisateurs, la question du ralliement au monde du libre n'est pas directement liée à la disponibilité des codes-sources. En revanche, le contexte d'évolutivité et de compatibilité des produits du libre les dégage d'une dépendance vis-à-vis du ou des éditeurs de logiciels, de la bonne volonté desquels ils dépendent, tant sur le plan de l'amélioration des versions disponibles que sur celui des possibilités de combinaison d'outils (interopérabilité). La question centrale pour eux se pose en termes de coût de migration par le fait que le passage d'une catégorie de standards à une autre pose le problème de la migration des produits d'application existants (données, textes, statistiques, objets formatés, programmes...), mais aussi de la remise en cause des externalités de réseau issues des possibilités d'échange entre utilisateurs partageant un même standard et enfin des efforts d'apprentissage consécutifs à l'adoption d'un nouveau standard.

Ces problèmes de migration recouvrent en réalité deux aspects complémentaires. En premier lieu, ils comprennent un aspect de dynamique, dans la mesure où des effets de rendements croissants d'adoption [Arthur, 1989] ont pour conséquence une diminution radicale des coûts de migration, au fur et à mesure du ralliement des utilisateurs. En revanche, pour les premiers utilisateurs qui décident de migrer vers le libre, il faut escompter que ces coûts de migration soient compensés soit par des avantages en termes de fonctionnalités d'utilisation, soit par une solide capacité d'anticipation et de croyance sur le devenir du libre. Évidemment, en deuxième lieu, cette difficulté peut se voir très fortement estompée par la possibilité de pouvoir passer d'un monde de compatibilité à un autre, à un coût faible et sans entraîner de trop fortes distorsions. Cette possibilité est conditionnelle de l'existence de

t e c hnologies passerelles (*gateway technologies*) dont l'objet est de permettre de faire passer un objet informationnel formaté en conformité avec un monde de compatibilité A vers un monde de compatibilité B et réciproquement. Les possibilités de pouvoir utiliser, dans un premier temps au moins, des logiciels libres dans un environnement Windows ou émuler un environnement Windows comme "bureau" ouvert sous Linux et dans le contexte d'une gestion multitâches, constituent bien entendu une avancée importante dans ce sens. De l'existence de telles passerelles dépendra la possibilité de constituer le monde du libre en alternative viable à celui d'un monde de production de logiciels fondé sur l'appropriation privée et le monopole d'exploitation des connaissances développées.

## Pourquoi le libre peut-il adopter les simples utilisateurs ?

Le logiciel libre est un phénomène initié et approprié par les professionnels de l'informatique, dans les centres de recherche (publics et privés) et dans les entreprises. Pourtant, dès le début, les informaticiens ont affiché la volonté que ces créations soient utilisées par tous. Cela, pour des raisons indirectes, politiques, philosophiques ou même psychologiques, mais aussi pour des raisons économiques.

### *Des justifications indirectes*

Pour certains développeurs, dans la lignée de Richard Stallman, le logiciel est un enjeu politique, de plus en plus important avec la place grandissante de l'informatique dans notre société. Comme le dit Georges[7] : "C'est quand même la première ressource, le premier produit qui ne soit pas en voie de privatisation mais en voie de socialisation. On est en train de privatiser l'eau, l'air bientôt, quand il sera pollué. Bon, là il y a un truc qui se crée, et on dit : *voilà, ça appartient à la société*". Et le logiciel libre peut être une base pour faire évoluer cette société, car, comme le souligne Yann : "Ce qui gère le monde d'aujourd'hui, c'est l'argent, il n'y a pas de secret, c'est l'économie [qui] ne peut pas se priver d'informatique. Tous les systèmes économiques reposent sur des systèmes informatiques, libérer ces systèmes-là, faire que ça tourne avec des logiciels libres, les gens vont réfléchir. Si l'on arrive à ne faire que du logiciel libre, les gens vont réfléchir."

C'est sans doute un tel argument qui permet de justifier l'engagement de développeurs, le démarrage de certains projets, même si, au niveau individuel, l'implication est toujours progressive et la contribution commence toujours autour de logiciels que l'on utilise.

Un autre argument théorique, avancé par Lerner et Tirole (2002) expliquerait l'intérêt des développeurs pour les projets à destination des simples utilisateurs par l'effet de réputation créé. Si l'on raisonne en terme de signal (ou d'ego d'ailleurs), il est effectivement plus valorisant de contrôler ou de

7. Extrait d'un entretien avec un développeur français. Déjà cité dans Demazière, Horn, Jullien 2004. Ce prénom et les suivants sont fictifs.

participer à un projet à destination du plus grand nombre. Cependant, les études empiriques ne valident pas toujours l'hypothèse d'une recherche de reconnaissance par les inventeurs de projets libres. L'analyse économique nous semble fournir des explications plus solides de la volonté du monde libre à intégrer les simples utilisateurs.

*Des raisons économiques pour prendre en compte les simples utilisateurs*

Comme nous l'avons expliqué dans la première partie, l'industrie informatique est une industrie à forts rendements croissants d'adoption, principalement, pour reprendre les catégories d'Arthur (1989), à cause des effets de réseau et des effets d'interrelation technologique.

Pour imposer sa solution, il faut donc convaincre le maximum d'utilisateurs et parmi eux, les simples utilisateurs.

*Les effets de réseau*

Le développement des réseaux dans l'organisation d'abord, puis entre organisations (Internet) ensuite, a relié différents groupes d'utilisateurs qui avaient peu de contacts auparavant. Aujourd'hui, l'ensemble des services d'une organisation échange *via* Internet des fichiers bureautiques (textes, feuilles de calcul, etc.). Les administrations publiques en dématérialisant certains actes (les déclarations d'impôt par exemple) s'adressent aussi à l'ensemble de la population, des utilisateurs les plus simples aux plus sophistiqués.

Il est alors important que les documents reçus ou les protocoles d'échange utilisés soient standardisés. La solution pour atteindre cette standardisation peut être la normalisation, mais elle est inefficace dans l'industrie du logiciel [Dréan, 1996 ; Zimmermann, 1995]. Le recours au marché laisse ces standards sous le contrôle d'un monopole, qui n'est pas forcément efficient. Le libre peut être en revanche une voie efficace pour garantir ce respect des standards [Zimmermann, 1999].

Pour les acteurs du libre, il s'agit donc d'imposer les outils de traitement de l'information, libres, qu'ils utilisent afin d'être sûrs de pouvoir continuer à les utiliser, dans une concurrence entre standards classiques. C'est d'autant plus vrai dans les logiciels métiers, où les logiciels portent souvent en eux les formats des documents qu'ils permettent de créer et il est difficile, coûteux, de créer les "technologies passerelles" entre les différents logiciels. Pour l'instant, les efforts des acteurs du libre ont surtout porté sur les logiciels de traitement de l'information "de base", c'est-à-dire communs (suite bureautique, navigateur, logiciel de courrier) car ce sont à la fois des logiciels que les développeurs utilisent, donc pour lesquels ils ont le plus d'incitations à voir une réalisation libre émerger, mais aussi ceux qui sont utilisés par le plus grand nombre, donc ceux pour lesquels les effets de réseaux sont potentiellement les plus forts.

De plus, ce sont aussi les logiciels que les dirigeants connaissent le mieux car ce sont ceux qu'ils utilisent. Ils sont un moyen de convaincre de la qualité des logiciels libres. Enfin, ils sont aussi nécessaires pour pouvoir

pousser à l'adoption des autres logiciels libres car ce sont souvent des logiciels qui fonctionnent, exploitent les données produites par d'autres logiciels (phénomène d'interrelation technologique).

*L'interrelation technologique*

Un logiciel ne fonctionne pas seul, mais grâce à d'autres logiciels et à un assemblage de composants matériels. L'adoption d'un type de composants a une influence sur l'adoption des composants liés. Par exemple, si Windows est si populaire, c'est parce qu'il existe de nombreux logiciels qui fonctionnent avec ce système d'exploitation. Et un développeur de solution a intérêt à développer son logiciel pour Windows car il s'adressera ainsi à la plus grande population potentielle d'utilisateurs. Cet effet auto-renforçant, l'interrelation technologique, explique la nécessité de développer des logiciels métiers ou grands publics. Ceux-ci sont nécessaires à l'adoption d'un quelconque système d'exploitation, libre ou non. Là aussi, les logiciels métiers, mais surtout les logiciels communs, apparaissent comme importants à développer. C'est d'autant plus vrai que les professionnels de l'informatique, et surtout les développeurs du libre viennent plutôt du "monde" Unix, tandis que les simples utilisateurs appartiennent plutôt au "monde" Windows [Jullien, 1999]. Ces logiciels communs ont été bien souvent, dès leur origine, développés sur plusieurs systèmes d'exploitation. Ces logiciels d'application agissent ainsi comme des passerelles entre deux architectures de systèmes incompatibles, facilitant ensuite le passage à une offre renouvelée en termes de système d'exploitation.

*Une conséquence : l'implication d'acteurs économiques majeurs du monde Unix dans ces développements*

Ces éléments expliquent que certains des logiciels libres grand public, les plus emblématiques (Open Office, Mozilla), aient été initiés par des entreprises (SUN, Netscape), le plus souvent issues du monde Unix. Les fabricants d'architectures de systèmes (SUN) ont besoin, pour les vendre, de développer des solutions d'usage, des applications, mais aussi de faciliter l'adoption de ces technologies, en abaissant le coût de changement pour des utilisateurs qui auraient déjà appris à utiliser les logiciels liés. Les fournisseurs de serveurs (Netscape) ou de services (AOL) ont besoin aussi de contrôler les logiciels clients pour être sûrs que leurs logiciels serveurs de contenu fonctionnent correctement avec eux. Nous pensons que l'arrivée de ces nouveaux utilisateurs peut au pire déstabiliser les groupes qui développent les logiciels et au mieux, ne rien changer dans la prise en compte de leurs besoins.

## Un compromis socioéconomique difficile avec les simples utilisateurs

Le compromis classique du logiciel libre, tel qu'analysé par von Hippel (2002), est celui de l'utilisateur-développeur : si un utilisateur veut de nouvelles fonctionnalités, c'est à lui de les amener dans le pot commun. Yann,

responsable d'un projet d'ERP libre, le résume ainsi : "Il y a des gens qui sont arrivés et qui ont dit voilà il faut que ça soit internationalisé, moi j'ai toujours dit si vous souhaitez le faire, faites-le, moi j'y consacrerais même pas une minute de mon temps parce que c'est pas quelque chose qui m'intéresse".

Les simples utilisateurs ne sont pas des développeurs, ne connaissent pas les règles qui organisent la production de tels logiciels. Ils sont potentiellement des éléments perturbateurs à une telle organisation de production et ont une vision déformée des avantages concurrentiels des logiciels libres.

### *Le point de vue des développeurs : des participants qui nuisent à la productivité*

L'intégration de ces utilisateurs dans le développement est potentiellement nuisible, car en plus de ne pas coopérer, ils peuvent diminuer la productivité du groupe par leurs questions ou leurs demandes. Pour continuer à citer le responsable du projet précédent, ces utilisateurs sont "des utilisateurs au vrai sens du terme. C'est-à-dire qu'ils n'ont de contact avec l'outil informatique qu'au travers d'un écran, une souris et un clavier. Ils ne vont pas plus loin, enfin on voit au niveau des questions, généralement les gens téléchargent [le logiciel], ils se retrouvent avec un tar.gz et ils posent la question à la liste : [mais qu'est-ce que je fais de mon tar.gz] ? Donc ils n'ont aucune compétence en informatique, ils sont vraiment là pour l'utilisation."

Comme ils ne sont pas non plus familiarisés avec les listes de diffusion, pour répondre à leur question il faut qu'il y ait "des gens qui soient là, qui soient d'une patience infinie et qui s'y mettent tous les jours et qui vont effectivement réexpliquer pour la quarantième fois qu'il faut faire telle et telle étape et que ceci est expliqué déjà quarante fois dans la liste". Ce comportement n'est pas le comportement des développeurs, et comme le dit notre interlocuteur, "[lui] il a toujours répondu [RTFM][8] au début dans le sens où [il n'était] pas là pour expliquer à quelqu'un comment installer MySQL".

D'autre part, les demandes de ces utilisateurs ne sont pas forcément en accord avec les demandes des développeurs[9] : les interfaces d'administration de Windows sont souvent assez simples pour permettre à un utilisateur sans connaissance de l'informatique de faire des tâches complexes d'un point de vue technique, comme installer une imprimante. Mais pour les développeurs, cette simplicité est un défaut, car les interfaces sont trop rigides et restrictives pour leur permettre de configurer réellement le système. Donc aller dans le sens des simples utilisateurs peut diminuer l'utilité des utilisateurs-développeurs.

En fait, ce risque d'une baisse de la productivité est limité par une série de filtres. Les utilisateurs ne sont pas directement en contact avec les développeurs, grâce à divers procédés techniques : existence de listes spéciales, privées, accès à l'espace de développement filtré. Les règles qui prévalent

8. Read the Fucking Manual, http://www.linux-france.org/prj/jargonf/R/RTFM.html et http://www.readthefuckingmanual.com/ pour une explication de l'intérêt de lire ces 'foutus manuels'.
9. L'exemple développé est extrait d'un article de John Gruber, "Ronco Spray-On Usability", disponible sur http://daringfireball.net/2004/04/spray_on_usability.

à l'utilisation des listes de diffusion et qui sont rappelées sur les sites des projets[10] , ainsi que les possibilités de rappel à l'ordre (RTFM), encadrent fortement la prise de parole. Mais le filtre le plus efficace est sans doute le filtre culturel : les discussions se font le plus souvent en anglais, les sites mêmes n'étant pas toujours traduits[11], avec des outils non maîtrisés par les simples utilisateurs (listes de discussions) et difficiles à trouver, même à l'aide d'un logiciel de recherche d'information comme Google. Cela d'autant plus qu'il n'est pas dans la culture des simples utilisateurs d'aller rechercher sur le Web des réponses à des problèmes informatiques.

*Le point de vue des simples utilisateurs : des logiciels gratuits*

L'analyse d'un développeur relative à l'adéquation d'un logiciel libre avec ses besoins se fera en terme de fonctionnalités (au sens de Lancaster), avec une estimation du coût de développement des fonctionnalités éventuellement manquantes. Pour les simples utilisateurs, entreprises (PME) sans compétence informatique ou particuliers, comme le résume Denis, un responsable d'une entreprise proposant des offres basées sur des logiciels libres, "quand vous êtes face à un client, la problématique de savoir s'il prend du libre, même s'il n'y comprend rien, ce n'est pas son problème. Son problème à lui, c'est le résultat. Si vous voulez, si c'est une entreprise industrielle en face, bon ben du libre, oui d'accord. Lui, ce qu'il va vous poser, c'est des questions hyper-traditionnelles : "combien ça coûte ?" et surtout "est-ce que vous êtes pérennes ou non ? ", "est-ce que vous êtes capable de m'accompagner dans le temps pour faire la presta ?" et aussi " en coût d'exploitation courant, combien ça va me coûter ?" Voilà, ce sont les questions de base, mais que vous fassiez du libre ou non, de toute manière ça va revenir de manière intrinsèque".

On est dans une concurrence entre produits, pas dans une analyse des avantages en terme de liberté, de facilité d'évolution future, etc. C'est même souvent la gratuité qui est perçue comme seul avantage concurrentiel du libre, comme l'explique Yann : "moi, c'est ce que j'ai rencontré honnêtement à chaque fois, chaque fois que j'ai entamé le processus de discussion avec les gens ben voilà votre besoin je peux vous le coder.

La réponse était toujours : oui ben non nous si on prend du libre c'est parce que ça coûte rien. "Comme le montre le rapport de Frédéric Labbé[12] ou la communication de la Gendarmerie[13], la réflexion sur le passage d'une solution basée sur MS Office à une bureautique libre n'est pas une réflexion en terme de fonctionnalité, mais en terme de coût (coût des licences et coût de gestion de ces licences).

10. Voir, par exemple, la page d'accueil des listes du projet OpenOffice : http://www.openoffice.org/mail_list.html
11. Toujours sur le projet OpenOffice, le site francophone et ses menus en anglais : http://fr.openoffice.org
12. Une des premières personnes à avoir effectué une migration d'une suite MSOffice à OpenOffice, dans un hôpital. http://www.framasoft.net/IMG/pdf/memoire_cnam.pdf
13. http://www.zdnet.fr/actualites/informatique/0,39040745,39203431,00.htm/

*Comportements opportunistes ou ignorance réciproque ?*

Ce comportement d'utilisateur pur peut être perçu comme un comportement de passager clandestin, dans la mesure où ces nouveaux utilisateurs bénéficient de produits d'une nature de bien public à la production desquels ils n'ont pas contribué. Mais le bénéfice qu'ils en retireront ne se faisant pas au détriment de producteurs, il n'est pas très nuisible à la coopération. Le vrai risque d'opportunisme est probablement plus du côté du potentiel d'affaires qui résulte de cette "marchandisation" du libre. Comme on l'a évoqué, certains agents sont susceptibles de tirer des bénéfices privés à partir de l'exploitation commerciale de produits au développement desquels ils n'ont pas nécessairement participé et qui n'ont donné lieu pour ce faire à aucune rémunération directe des contributeurs[14].

On est ainsi conduit à la confrontation entre deux mondes aux fonctionnements différents qui ne peut être sans conséquence. Bénévoles et peu gourmands en termes de rémunération, les développeurs du libre se voient interagir de manière régulière avec des entrepreneurs dont les exigences en matière de revenus, mais aussi de délais, de pression pour sortir des fonctionnalités même si elle ne sont pas forcément stabilisées... sont d'un autre ordre. Parfois le développeur se fait entrepreneur, parfois il se contente de fournir à celui-ci les produits à la base de son activité, dans tous les cas s'instaure un écart qui, s'il affecte les rythmes et mode de vie, affecte aussi de manière visible les niveaux de revenu. Cet état de fait est susceptible de remettre en cause la structure des motivations à contribuer des développeurs et, remettant en avant les dimensions utilitaristes de ces motivations, il risque de décourager la mobilisation des développeurs et donc de nuire à l'efficacité du processus coopératif. Il peut entraîner la sortie de certains des contributeurs comme résultat d'une frustration engendrée par cette confrontation [Foray et Zimmermann, 2001].

Il peut aussi remettre en question les modalités d'allocation de ressources qu'effectue chaque contributeur individuel, lequel peut être tenté de consacrer tout ou partie de son temps à des activités plus lucratives, en relation ou non avec la sphère du logiciel libre, ou en tout cas plus tourné vers la satisfaction de ses besoins propres en terme de fonctionnalité du logiciel.

Cette conciliation difficile entre logiques propres des simples utilisateurs et des développeurs risque d'avoir des conséquences en termes de variété des produits développés en libre. Il ne faut pas oublier en effet que la motivation à l'origine du développement du libre reste avant tout la production de code en réponse aux propres besoins des programmeurs. Cet effort collectif d'amélioration d'une base de logiciels qui engendre pour chacun une externalité d'usage ne suit donc aucunement une logique de marché. Ainsi, on peut constater de réelles difficultés à mobiliser des programmeurs en libre pour le développement de logiciels libres métier (cf. les difficultés

14. Nous entendons ici que l'organisation productive du libre ne rémunère pas les développeurs, ce qui n'exclut pas que ceux-ci puissent être éventuellement rémunérés par d'autres organismes, privés ou publics.

d'émergence des ERP libres), pour lequel le retour des utilisateurs est important et l'intérêt technique faible (donc peu de motivations pour les développeurs pour "s'y mettre"). Plus encore il y aurait lieu, de déployer, dans les groupes libres, de nouvelles compétences pour convaincre les utilisateurs, car il faut passer d'une stratégie *consumer pull* (ce sont les informaticiens qui vont tester le logiciel disponible sur un site et vont l'adopter) à une stratégie *product push* (marketing, publicité, offre d'installation, etc.) donc où les acteurs marchands vont avoir une place importante. Il est à ce sujet révélateur que, pour la première fois dans un projet libre, OpenOffice ait un sous-projet chargé du marketing du produit[15].

Pourtant, malgré ces contradictions apparentes, les projets de logiciels libres pour le grand public et dans les applications métier se développent, et il semble qu'ils soient de plus en plus populaires. On peut essayer de définir, toujours en étudiant les exemples de diffusion, une nouvelle écologie du libre, où les simples "utilisateurs" contribueraient, directement ou indirectement à la production coopérative.

## Y a-t-il de nouvelles catégories d'incitations ?

Comment les simples utilisateurs peuvent-ils favoriser le développement des logiciels (libres) sans que la marchandisation que cela entraîne n'éloigne les utilisateurs-développeurs ? Comment, d'un autre côté, les besoins fonctionnels de ces utilisateurs pourraient-ils être pris en compte, notamment en ce qui concerne les logiciels métiers ou des logiciels de traitement de l'information[16] ? La deuxième partie du dilemme est sans doute plus facile à résoudre que la première.

### *Des logiciels libres pour les non-informaticiens*

Si l'on traite des logiciels professionnels (logiciels de comptabilité, logiciel médical, etc.), on peut estimer que le développement libre est possible, sur le même compromis social que les logiciels professionnels pour les informaticiens. En effet, pour beaucoup de ces logiciels, il existe suffisamment de professionnels ayant des compétences en informatique pour qu'une population suffisante produise une offre libre : la diffusion d'Internet, comme elle a permis le rapprochement des informaticiens, permet aussi le rapprochement de ces professionnels. Par exemple, l'ERP libre Dolibarr a été développé par un créateur d'entreprise, informaticien, et la suite médicale MedinTux par un médecin urgentiste[17].

Là encore, l'effet réseau créé par Internet explique ces développements, offrant à des compétences rares (utilisateur-développeur et professionnel) la possibilité de s'identifier *via* les moteurs de recherche et de collaborer. Donc une écologie libre peut se mettre en place autour de nombreux

15. http://marketing.openoffice.org/
16. *User as innovator.*
17. http://upsis.club.fr/MedinTux/. Ce logiciel a été lauréat des derniers "trophées du libre 2005".

besoins fonctionnels, hors de la sphère professionnelle de l'informatique, mais elle est alors, une fois encore, fondée sur l'expression des besoins des utilisateurs-développeurs. Et se repose la question du comment intégrer les besoins des simples utilisateurs, importante pour la viabilité à long terme du modèle et pour sa capacité à former une alternative crédible au modèle propriétaire.

### *Accueillir les simples utilisateurs pour renforcer la dynamique d'un projet*

Tout d'abord, on peut considérer que l'adoption de logiciels libres par une population sans cesse élargie de simples utilisateurs constitue un motif de satisfaction pour la communauté du libre : le niveau d'adoption des logiciels libres témoigne de ce que leur qualité intrinsèque incite certains utilisateurs à sortir d'une situation antérieure d'usage de logiciels propriétaires. Elle renforce corrélativement le poids de la voie *open source* face au modèle propriétaire et donc les conditions de sa viabilité et la concurrence dans l'industrie du logiciel, notamment face à la position dominante de Microsoft. En un sens, beaucoup de contributeurs du libre sont explicitement ou non des adversaires acharnés de Microsoft et ne renâclent pas devant un certain prosélytisme.

### *La diversité des rôles dans le logiciel libre laisse une place pour les simples utilisateurs*

Paradoxalement, l'arrivée des simples utilisateurs permet aussi d'augmenter l'efficacité des projets libres en augmentant les compétences disponibles. En effet (figure 1), les contributions à un projet libre ne se limitent pas au développement du code souvent réalisé d'ailleurs par un petit groupe.

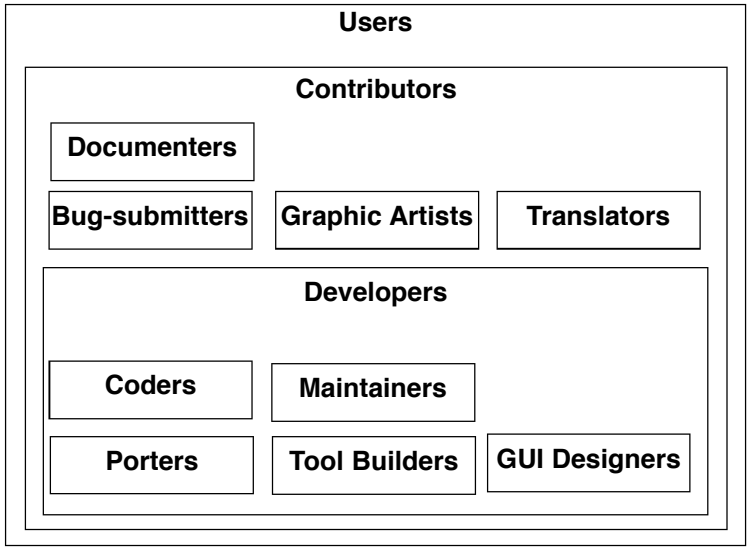



***Figure 1. La multiplicité des rôles dans le modèle open source (OSS Watch http://www.oss-watch.ac.uk/)***

Les interfaces graphiques, les traductions, appellent de nouvelles compétences, nécessaires à la diffusion vers ces nouveaux publics, mais non possédées par les utilisateurs-développeurs. Un second aspect positif à ce niveau collectif tient au fait que les nouveaux utilisateurs constituent un potentiel de contribution au modèle de développement du logiciel libre, en tant que base de test et source d'amélioration des logiciels.

Ainsi une communauté *open source* ne se limite pas aux seuls développeurs, mais prend aussi appui sur une catégorie "d'utilisateurs-frontière" [Kogut and Metiu, 2001] qui ne sont pas capables de contribuer au développement du logiciel, mais ont un rôle essentiel à jouer en termes de source de l'innovation [Von Hippel, 1988] et surtout en tant que base de test et de mise au point. "La source de cet avantage tient à la simultanéité du développement et de la mise au point. Malgré son aspect peu attractif, la maintenance à elle seule représente toujours 50 à 80 % du coût d'un logiciel. La plus grande part de la communauté n'est pas impliquée dans l'écriture du code mais dans son débogage" [Kogut and Metiu, 2001].

Le seul point d'interrogation relatif à cet aspect est celui de l'organisation et de l'efficacité de la collecte des informations pertinentes à faire remonter de la base des utilisateurs vers la communauté des développeurs. Des solutions techniques permettent de faciliter ces retours : OpenOffice intègre un système de reporting automatique des erreurs, demandant simplement aux utilisateurs d'indiquer ce qu'ils étaient en train de faire lorsque le logiciel a planté.

### *Une source de rémunération pour les entreprises du libre et pour les utilisateurs développeurs*

Mais la responsabilité de l'interface entre ce monde de simples utilisateurs et le cœur de la communauté des développeurs échoit aussi naturellement aux entreprises qui commercialisent les produits du libre et des services d'aide à l'utilisation en direction de ce marché de non-spécialistes. Elles peuvent former le cœur d'un mécanisme efficace de collecte et de remontée de l'information. C'était déjà l'argument d'Ousterhout (1999) pour dire que le "*free software needs profit*". Ainsi, le développement des sociétés d'installation et de maintenance des ordinateurs en direction des particuliers, ainsi que l'augmentation de la complexité des équipements à domicile (ADSL, multiples ordinateurs, souvent en réseau), favoriseront sans doute ces retours.

En outre, les arguments d'efficacité collective à laquelle les entreprises commerciales impliquées dans la distribution de libre se doivent de contribuer, se doublent d'une capacité à mobiliser certains développeurs de bon niveau sur la base d'un ordre plus utilitariste de motivations.

Dans Foray, Thoron et Zimmermann (2006) nous montrons, sur la base d'un modèle de différenciation verticale dans l'offre de services en direction de simples utilisateurs, que la perspective de gains commerciaux peut s'inscrire en concomitance avec une contribution accrue de la part des développeurs les plus expérimentés. Ainsi, les perspectives liées à la marchandisation

du libre ne conduisent pas nécessairement à des comportements opportunistes mais jouent au contraire, pour les plus compétents, un rôle incitatif complémentaire aux effets d'apprentissage pour les moins experts[18].

Certes, il serait naïf de considérer que ce résultat devrait se traduire dans la réalité par un engouement généralisé de la part des informaticiens chevronnés pour la création d'entreprise. Von Hippel (2002) relève que les membres de la communauté du libre sont en général peu enclins à investir temps et argent dans une aventure entrepreneuriale. On peut donc concevoir que ce type d'incitation ne jouera en direct que pour un petit nombre d'entre eux et se traduira plus souvent par l'embauche d'informaticiens de haut niveau dans des entreprises qui trouvent un intérêt économique effectif à s'impliquer dans les deux mondes, du marchand et du non-marchand. Ainsi Lakhani et Wolf (2005) relèvent-ils que : "*a majority of our respondents are skilled and experienced professionals working in IT-related jobs, with approximately 40 percent being paid to participate in the F/OSS project*."

Finalement, les simples utilisateurs ne financeraient pas directement (ou au moins consciemment) des contributions au logiciel, mais un service de maintien de capacités techniques, de fonctionnalités du logiciel et d'interfaçage de ces capacités avec leurs besoins. Et c'est pour répondre à cette demande que les fournisseurs de services contribueraient au développement des logiciels libres.

Il nous faut, pour conclure, signaler que cette évolution risque encore d'accentuer les tensions entre les responsables de projets, promoteurs d'une certaine vision du Libre et ces entreprises et utilisateurs, beaucoup plus demandeurs d'une réponse rapide à une question (bogue, nouvelle fonctionnalité, etc.) Ces nouveaux utilisateurs s'ils en ont la capacité financière, peuvent avoir un impact majeur sur l'orientation, les objectifs des projets libres car les contraintes industrielles ne sont pas forcément les mêmes que les contraintes des "artisans", amoureux de la belle ouvrage à l'origine d'un projet.

Ainsi, l'objectif principal de Debian était d'être la distribution GNU/Linux la plus stable car la plus testée, au détriment de l'ajout des nouvelles fonctionnalités. Le projet Ubuntu[19] est basé sur Debian, mais ses engagements sont sur le fait d'être "*released regularly and predictably; a new release is made every six months. Each release is supported with free security updates and fixes for at least 18 months.*"

Cette distribution propose aussi les dernières versions des logiciels (par exemple, la version 2 d'OpenOffice, qui n'était pas encore stable quand la "*release*" la proposant est sortie) et elle apparaît comme concurrente à Debian dans la communauté[20]. Dans le cas du projet SPIP, des administrations

18. "The differences between the two groups (of contributors) are consistent with the roles and requirements of the two types of F/OSS participants. Paid contributors are strongly motivated by work-related user need (56%) and value professional status (22.8%) more than volunteers. On the other hand, volunteers are more likely to participate because they are trying to improve their skills (45.8%) or need the software for non-work purposes (37%)." (Lakhani and Wolf, 2004)

19. http://www.ubuntulinux.org/

20. cf. l'interview de l'initiateur d'Ubutu, Mark Shuttleworth http://www.framasoft.net/article624.html

françaises ont "demandé" le développement de fonctionnalités incompatibles avec les objectifs initiaux du projet, créant un nouveau projet SPIP Agora, concurrent du projet initial[21].

Ces deux exemples indiquent que la demande en terme de fonctionnalités exerce une pression sur le cycle de développement d'un projet libre, cycle d'autant plus difficile à tenir que les contraintes commerciales sont fortes. Est-ce au détriment de la qualité des logiciels développés ? Sans doute en termes de stabilité et de cohérence des logiciels, pas forcément en termes de richesse fonctionnelle, donc de qualité ou de performance perçue par l'utilisateur final : par exemple, un navigateur qui n'accepterait pas de *plug-in* ne permettrait pas d'ouvrir certain sites Web, même s'il peut être considéré comme plus sûr.

Le succès des premiers logiciels libres est dû à leur capacité à proposer une base technique de qualité. Sans doute les futurs logiciels à succès seront-ils ceux qui, sur cette base solide, permettront le développement de composants complémentaires permettant de développer les fonctionnalités du produit, de changer son ergonomie, etc[22].

21. Voir l'article de Demazière, Horn et Zune pour une description approfondie de ce cas
22. C'est par exemple ce que propose Mozilla https://addons.mozilla.org/?application=firefox

# BIBLIOGRAPHIE


ARTHUR W. B., "Competing technologies, increasing returns and lock-in by historical events : The dynamics of allocations under increasing returns to scale". *Economic Journal*, 99 : 116-131, 1989. URL: http://www.santafe.edu/arthur/Papers/Pdf_files/EJ.pdf.

DEMAZIÈRE D., HORN F, AND JULLIEN N., "Le travail des développeurs de logiciels libres. La mobilisation dans des "communautés distantes". Conf. Proximity, Networks and Co-ordination, Marseille, June 17th-18th, 2004. À paraître, *Cahiers Lillois d'Économie et de Sociologie*.

DRÉAN G., *L'industrie informatique, structure, économie, perspectives*. Masson, Paris, 1996.

FORAY D. ET ZIMMERMANN J.B., "L'économie du logiciel libre : organisation coopérative et incitation à l'innovation", *Revue Economique,* vol. 51, Octobre pp.77-93, 2001.

FORAY D., THORON S. ET ZIMMERMANN J.B., "Open Software : Knowledge Openness and Cooperation in Cyberspace", à paraître in E.Brousseau and N.Curien (Eds.), *Economics of Internet*, Cambridge University Press, 2006.

KOGUT B. AND METIU A., "Open Source Software development and Distributed Innovation", Reginald H.Jones Center Working Paper #01-08, April 2001.

JULLIEN N., "Linux : la convergence du monde Unix et du monde PC". *Terminal*, 80/81:-70, 1999.

JULLIEN N., "Le marché francophone du logiciel libre". *Systèmes d'Information et Management*, n°1-Vol 8, 2003, pp. 77-99, 2003.

LAKHANI K. ET VON HIPPEL E., "How open source software works : "Free" user-to-user assistance", MIT Sloan School of Management, Working Paper #4117, 2000.

LAKHANI ET WOLF, "Why Hackers Do What They Do : Understanding Motivation and Effort in Free/Open Source Software Projects", in Feller, J., R. Fitzgerald, S. Hissam, & R. K. Lakhani (Eds.). *Perspectives on free and open source software* , MIT Press, 2005.

LERNER J. ET TIROLE J., "Some simple economics of open source software", *The Journal of Industrial Economics*, Vol.L, N°2, June, 197-234, 2002.

MAUSS M., "Essai sur le don. Forme et raison de l'échange dans les sociétés archaïques", *L'Année Sociologique*, 1923-1924.

OUSTERHOUT J., "Free software needs profit". Communications of the ACM, avril, 42(4) :-45, 1999.

VON HIPPEL E., "The Sources of Innovation", Oxford University Press, 1988.

VON HIPPEL E., "Open Source Software as horizontal innovation networks – by and for users", MIT Sloan School of Management W.P. N°4366-02, 2002.

ZIMMERMANN J.B., "L'industrie du logiciel, de la protection à la normalisation", in J.Perrin et al. (Eds.) *Changement institutionnel et changement technologique*, CNRS Editions, 1995.

ZIMMERMANN J.-B. "Logiciel et propriété intellectuelle : du copyright au copyleft". *Terminal* 80/81: 95-116. Special Issue, Le logiciel libre, 1999.