

LES LOGICIELS LIBRES ET GNU/LINUX

Marc SCHAEFER

Programme

- Le logiciel libre
- Les logiciels propriétaires
- Les problèmes
- La sécurité
- Ethique et responsabilité
- GNU/Linux
- Les acteurs du marché
- Les distributions
- Conclusion
- Questions

Motivations

- la maîtrise de l’outil informatique implique un contrôle accru sur celui-ci
- l’informatique nous confronte à des problèmes éthiques en plus des problèmes techniques !
- les logiciels libres sont de plus en plus utilisés sur les serveurs, mais aussi sur le poste de travail à terme.

Motivations (2)

- besoin en information
- aspects éthiques
- aspects médiatiques
- problèmes de sécurité, virus, fiabilité
- coût/impact à long terme
- *autres ?*

vos motivations pour être ici

Buts

- montrer les problèmes posés par les logiciels propriétaires
- montrer ce qu'apportent les logiciels libres (standards, interopérabilité, choix)
- donner quelques pistes pour une transition éventuelle

Le logiciel libre

Le logiciel libre est un logiciel qui peut être utilisé, copié, distribué, étudié, modifié et amélioré, par tous ses utilisateurs sans restrictions, ceci transitivement. De plus, toute garantie de la part de l'auteur ou de l'éditeur sont explicitement exclues.

Cela n'implique pas que le logiciel soit gratuit ou non-commercial !

partage, mise en commun, communauté, société

Les logiciels propriétaires

- relative complexité des logiciels
- courses aux fonctionnalités (*features*) en préjudice à la simplicité et fiabilité / sécurité
- viabilité sujette à caution
 - disparition / rachat / fusion / intégration
 - mises à jour forcées
- obsolescence induite du matériel
- absence de compatibilité entre plateformes

rendent l'utilisateur *captif*

Les problèmes

- éthique des licences et achats de logiciels
- mises à jour
- virus, worm, piratages, non fiabilité (sécurité)
- pérennité de vos données (formats)
- formation des utilisateurs souvent insuffisante

La sécurité

Pourquoi a-t-on autant de problèmes aujourd'hui ?

Il n'y avait pas autant de problèmes de sécurité avant la mise sur Internet ! mettons un firewall ! est-ce la vérité ? est-ce suffisant ?

Le modèle économique de l'industrie propriétaire du logiciel est-il simplement incompatible avec la sécurité ?

L'ouverture d'Internet et la relative bonne connaissance des standards liés à TCP/IP est-il un plus ou un moins ?

La sécurité : pourquoi autant de problèmes

- les auteurs de logiciels propriétaires considèrent souvent la sécurité comme un problème de relation publiques plutôt qu'un problème technique
- différence marketing entre produits utilisateurs et pro et entreprise alors que les besoins de sécurité sont identiques
- les utilisateurs se contentent de solutions partielles (anti-virus p.ex.)
- on considère souvent, à tort, que la menace vient uniquement de l'extérieur : les réseaux internes sont souvent mal protégés

- on oublie d'intégrer la sécurité à la conception de logiciels, au déploiement de systèmes et à la formation
- on néglige parfois des solutions simples pour des solutions accrocheuses plus complexes

La sécurité : espoir avec les logiciels libres

- grand choix et adaptabilité
- comparaison sur des critères techniques grâce à une relative transparence, au moins pour les techniciens
- meilleure formation des utilisateurs (car nouveau système)
- formats de données ouverts et plus simples
- uniformité verticale et horizontale des logiciels
- modèle économique privilégie les services (et non les produits et la vente forcée de mises à jour de fonctionnalité)

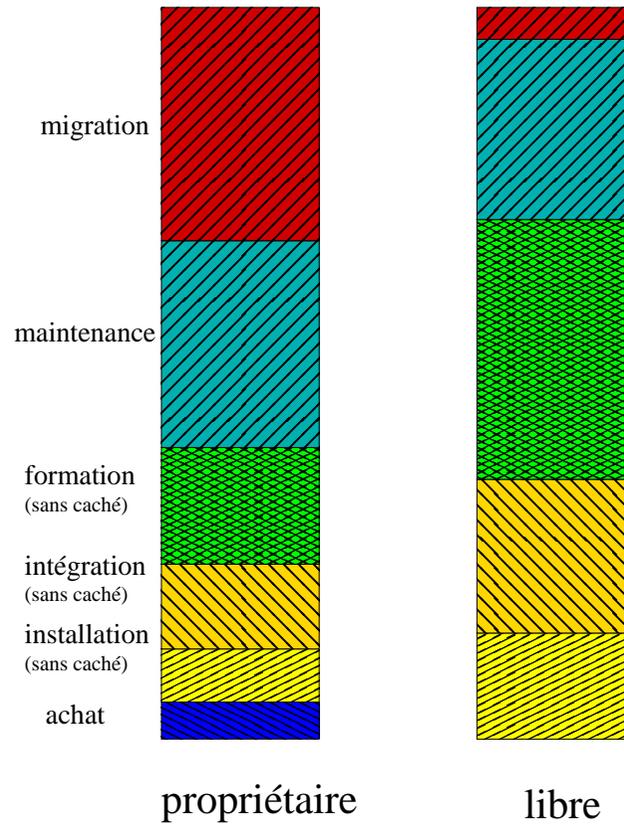
Ethique et responsabilité

- constat : les particuliers (en majorité) et certaines entreprises copient et utilisent illégalement de nombreux logiciels
 - aujourd'hui la connaissance de certains logiciels propriétaires est nécessaire pour obtenir un emploi, le coût théorique de l'acquisition de cette connaissance est élevé
 - à qui appartiennent vos données ? certains *formats de données* sont liés à une application particulière (*fermés*). Certains éditeurs font attention à utiliser des formats ouverts, d'autres non
- vos données devraient vous appartenir : en pratique est-ce le cas ? avez-vous vraiment la possibilité de changer de logiciel ?

Ethique et responsabilité (2)

- dans quelle mesure peut-on être espionné ?
- dans quelle mesure le choix va-t-il s'amenuiser pour le consommateur et l'entreprise ?
- quand un monopole est-il dommageable ?
- il est responsable d'assurer un revenu à ceux qui écrivent ou maintiennent et supportent un logiciel
- le système des brevets a montré ses limites – ou celles des structures

Le coût du logiciel



Exemple de comparaison de coût Bien sûr, ce coût est très variable. Dans certains cas (p.ex. logiciels bureautique propriétaire), les coûts d'installation, d'intégration et de formation sont souvent considérés comme nuls (ils sont payés à l'achat de la machine et lors de la formation gratuite aux logiciels propriétaires dans les Ecoles publiques p.ex.).

Le cas présenté reflète plutôt l'utilisation en entreprise d'un logiciel complexe (**Total Cost of Ownership, TCO**).

La Confédération propose un outil d'évaluation open source, <http://www.tcotool.org/>

Coût cachés à comparer Les coûts cachés (maintenance (pannes, mises à jour, incompatibilités), ainsi que la migration à un autre logiciel lorsque celui-ci n'existe plus) forment la partie la plus importante en logiciel propriétaire. Notamment la maintenance voit son coût augmenter plus le logiciel est utilisé longtemps. La migration est parfois très difficile d'un logiciel propriétaire à un autre logiciel (propriétaire ou non).

Résumé A court terme, le logiciel libre coûte probablement autant que ses équivalents propriétaires. Cela infirme donc l'idée préconçue que le logiciel libre est forcément gratuit.

Le coût est cependant réparti différemment : un coût de démarrage élevé (problématique de la formation gratuite sur la plateforme propriétaire ; problème de la croyance que des logiciels modernes ne nécessitent pas de formation ; également dû à la nature des logiciels libres, souvent conçus pour résoudre un problème technique donné, alors que le logiciel propriétaire est plus souvent conçu comme un produit fini correspondant aux besoins d'un panel de test représentatif des consommateurs)

A long terme, le logiciel libre propose des mécanismes qui peuvent rendre les coûts de maintenance et de migration plus faibles, notamment en permettant des économies induites par une meilleure formation, elle même rendue possible par transfert de charges.

Licences logicielles

une licence :

- régit les rapports entre *auteur* et *utilisateur* (sens large)
- base légale : *droit d'auteur*
- certains droits *inaliénables*
- autrement, selon la licence
- comment déposer son logiciel ?

Licences Tous les logiciels vendus et diffusés sont accompagnés d'une licence. Celle-ci est assimilée à une *permission* et précise les conditions d'utilisation du logiciel. Seule la mise dans le domaine public évite une licence : mais alors aucune protection n'est offerte pour l'auteur. Même les licences de logiciels libres utilisent les licences, parfois afin d'assurer la pérennité des droits pour tous les utilisateurs (*copyleft* au lieu de *copyright*).

Les licences dites copyleft Ces licences assurent que le logiciel ne pourra pas être rendu propriétaire. L'utilisateur final peut exiger, en vertu de la licence, d'obtenir le code source du logiciel original et des modifications, de la personne de laquelle il a reçu le logiciel, ceci transitivement jusqu'à l'auteur original.

Logiciel propriétaire : auteur \rightarrow auteur \rightarrow utilisateur

Logiciel libre : auteur \leftarrow auteur \leftarrow utilisateur

Ce sont bien les droits de l'utilisateur final qui priment dans ce cas !

(attention : en pratique, bien souvent c'est l'auteur de niveau supérieur qui peut agir

légalement, en particulier si le logiciel libre a été camouflé et la licence libre non transmise à l'utilisateur final !)

Droits inaliénables Les droits suivants ne peuvent être refusés à l'utilisateur d'un logiciel en Suisse (les pays européens ont une législation similaire) :

- droit d'examiner et de tester le programme, d'en contrôler le fonctionnement dans le but de rechercher les idées et principes (*droit européen*)
- décompilation pour compatibilité, mais uniquement pour interopérabilité (remplacement du programme exclu)

En pratique le droit à la décompilation est inapplicable : renouvellement rapide des logiciels, informations difficilement accessibles, coût d'une décompilation.

Brevets Pour les logiciels, il n'y a aucune protection des idées abstraites, seule l'expression concrète et l'algorithme sont protégés.

Les brevets ne sont pas pour l'instant applicables au logiciel en Europe. Malgré tout, certains procédés techniques relatifs au logiciel ont été acceptés comme brevets (30'000 environ). Une des compagnies au monde avec le plus de brevet est IBM,

qui les utilise – pour le moment – plutôt comme défense lors de procès : mon brevet contre ton brevet.

Informations sur les brevets : <http://swpat.ffii.org/>.

Valeur légale D'après la Convention de Berne, il n'est même pas nécessaire que la mention de copyright existe pour que le droit d'auteur soit applicable. Mais encore faut-il prouver que l'on a écrit ce logiciel ! Certaines associations offrent des services (payants) de dépôt de logiciels (ou de signatures) électroniques ou physiques (**APP, Interdeposit**), de notariat virtuel. Sinon reste bien sûr la diffusion open source (vous trouverez toujours quelqu'un qui a utilisé votre logiciel depuis le début), ou l'envoi à soi-même d'une enveloppe fermée recommandée.

Les avantages du libre (résumé)

- standards ouverts
- multiplateforme
- sécurité
- adaptable
- compatibilité
- indépendance, partage
- rentabilise les développements internes

mais ...

Les standards ouverts garantissent la pérennité des choix stratégiques et données. Le support multiplateforme garantit la pérennité de l'investissement matériel. La sécurité est vérifiable grâce aux audits indépendants. Le logiciel est très adaptable aux problèmes rencontrés. Le libre est compatible avec l'existant UNIX, Windows et MacOS. Il est indépendant de fournisseurs ou de pays. Il rentabilise les 80% de développement à façon interne (p.ex. projet Matra).

Synthèse des différences

- **TCO** différent
- différence des licences
- adéquation différente
- support obtenu différemment
- responsabilisation

Le *Total Cost Ownership* est réparti différemment : le coût du logiciel libre est général nul à l'achat. Sa mise en oeuvre par contre coûte en général très cher. Si vous chargez quelqu'un dans votre entreprise ou à l'extérieur de modifier ce logiciel, un coût supplémentaire existe. De plus, un coût de formation existe. En contrepartie, le logiciel pourra correspondre nettement plus à vos besoins et même évoluer avec vous ou votre entreprise : de plus des mises en oeuvres ultérieures ne sont pas nécessaires : seules des adaptations incrémentales sont en général nécessaires. Enfin, ce coût diminue proportionnellement au nombre d'installations.

Mais n'oublions pas que le coût d'un logiciel propriétaire n'est pas nul même après l'achat, en particulier si votre propre personnel ne peut jamais résoudre les problèmes lui-même !

Modèle économique

- basé sur les services
 - création/adaptation de logiciels
 - documentation, traduction
 - intégration
 - installation, mise en place, déploiement
 - formation
 - support technique
- tissu économique *local*

Qui utilise le logiciel libre ?

- GARTNER, mars 2004 : entre 70 et 80% des entreprises ont déployé un système GNU/Linux sur leur réseau
- 56% des grands groupes mondiaux (*Forester Research*)
- 90% des serveurs de noms d'Internet aux USA
- 70% des serveurs de courrier électronique
- 60% des serveurs WWW
- IBM, HP, Sun, UBS, SAP, Education nationale française, gouvernement allemand, chinois, mexicains, etc.
- VD, NE, Confédération, TF, Uni/EPF/HES

Qu'est-ce que GNU/Linux ?

- un OS gratuit (surtout libre, en fait)
- un système UNIXTM-compatible
- une implémentation libre (GPL) de la norme POSIX
- un kernel, ou un kernel plus un environnement d'exécution, des utilitaires, etc.
- un système **multitâche préemptif, multi-utilisateur, multiplateforme**, à mémoire **virtuelle et protégée**, suivant des normes et **standards ouverts**, à philosophie **UNIX**.

GNU/Linux

- filiation UNIX : 1969
- un kernel : Linux (août 1991)
- l'environnement de développement GNU
- la bibliothèque C standard GNU
- les utilitaires de base GNU, BSD et autres
- des logiciels *packagés* par chaque *distribution*

Les acteurs du marché

- les développeurs
- les distributions
- les conseillers
- les organisations de support
- les formateurs
- les associations
- les super-utilisateurs
- les utilisateurs

Les distributions

- collectent du logiciel libre sur Internet, le testent, l'intègrent (packaging) et le distribuent de manière cohérente (packages compatibles ensemble)
- différences entre les distributions
 - techniques
 - nombre de plateformes supportées (y compris systèmes embarqués p.ex.)
 - format de packaging
 - logiciels spécifiques de gestion

Les distributions (2)

- différences entre les distributions (suite)
 - organisationnelles
 - débats/bugs ouverts (publics) ou non
 - qualité de développement, test, assurance
 - type de cycle de vie et longueur du support et disponibilité des mises à jour
 - inclusion de logiciel propriétaire éventuelle
 - politiques
 - contrat social ou non
 - liens ou non avec entreprise donnée

Exemple de contraintes 1

- serveur de téléphonie vendu comme solution complète
- contraintes
 - support architecture IA32 et ARM
 - stabilité des changements logiciel, garantie de disponibilité des mises à jour pendant 3 ans
 - légèreté de la distribution, absence d'outils complexes de gestion
 - absence de logiciel propriétaire assuré

Suggestion ici : Debian GNU/Linux stable

Exemple de contraintes 2

- station de travail avec changement fréquent du matériel
- contraintes
 - support architecture IA64
 - besoin de mettre à jour les logiciels de façon fréquente, de manière à être compatible avec les derniers périphériques
 - pas de fonctionnement 24x7 à garantir
 - gestion simplifiée

Suggestion ici : Ubuntu GNU/Linux stable (basé sur Debian)

La formation

- place importante
 - la formation des utilisateurs (sécurité, etc) est souvent incomplète
 - les logiciels libres, pour déployer leur potentiel et éviter les pièges ont besoin de formation
 - les logiciels libres ne bénéficient pas encore d'une offre de formation de base gratuite via la formation générale ou les pré-requis

Avant tout : une démarche

le logiciel libre est avant tout une démarche, un choix, basé sur divers critères (éthiques, techniques, politiques, etc)

individuelle et *communautaire* à la fois !

Migrer en logiciels libres

- faut-il passer à GNU/Linux obligatoirement ?
- peut-on migrer facilement ?
- les solutions sont-elles matures, *compatibles* ?
- comment me former ? où trouver du support ?
- est-ce un investissement intéressant à long terme ?

Utilisation partielle du logiciel libre

- l'argent du beurre sans le beurre : profiter des standards ouverts en restant propriétaire : logiciels libres sous Microsoft Windows ou Apple MacOS X
- faire jouer la comparaison et la concurrence
- commencer peut-être par l'ingénierie et les serveurs, pas par les postes clients visibles

Mais : le logiciel libre reste une *démarche* !

Le logiciel libre apporte certains acquis : dans certains cas le logiciel propriétaire amène certains de ces acquis. Choisir un logiciel reposant sur un standard ouvert est le premier pas dans la bonne direction !

Le plein potentiel du logiciel libre se développe vraisemblablement plus facilement dans une entreprise d'informatique qui utilise les technologies Internet, ou fait du développement de logiciels, par exemple.

Mais mettre en oeuvre le logiciel libre est avant tout une démarche, qu'elle soit économique, de développement durable, ou sociale. En ce sens, c'est une nouveauté en informatique. Mais bien d'autres domaines, par exemple la gestion financière, ont mis du temps avant de développer des produits éthiquement défendables.

L'avantage est que même une utilisation partielle du logiciel libre ou open-source va mener à une pression sur les fournisseurs de logiciels fermés.

Malheureusement cela veut dire beaucoup d'énergie perdue pour la compatibilité avec les environnements et applications des environnements propriétaires, et finale-

ment une contribution à maintenir le modèle propriétaire comme standard de référence et plateforme universelle.

Conclusion

- les logiciels libres sont une alternative intéressante aux logiciels propriétaires par les possibilités techniques qu'ils offrent mais aussi par leurs caractéristiques non techniques
- on peut dès aujourd'hui utiliser les LL sur une plateforme propriétaire
- la philosophie des LL transfère une partie du coût d'un logiciel dans le support et la formation
- vous pouvez vous renseigner sur les logiciels libres disponibles, l'impact de l'informatique propriétaire sur votre vie privée et celle de la cité, voire commencer dès aujourd'hui à utiliser des logiciels libres

Plus d'information

- LIVRET DU LIBRE
- Rapport au gouvernement anglais concernant *GNU/Linux sur le poste de travail*, traduction française complétée à :
<http://www.iful.org/solutions/pdt/rapport-pdt>
- Le Groupe romand des Utilisateurs des Logiciels Libres :
<http://www.linux-gull.ch>
- Free Software Foundation Europe :
<http://www.fsfeurope.org>
- Linux : <http://www.linux.org> et <http://www.li.org>
http://en.wikipedia.org/wiki/Free_software
Wikipedia : logiciel libre

Vos questions ou remarques

Avez-vous des questions ou remarques ?

Questions

- échanger des données avec des logiciels Microsoft Office ?
- remplacer un serveur Microsoft Windows NT, Novell, ou Apple MacOS ?
- porter facilement un programme Microsoft Windows sous Linux ?
- tourner des programmes Microsoft Windows sous Linux ?
- remplacer un serveur UNIX par Linux ?
- tourner des logiciels propriétaires sous Linux ?
- remplacer MS-Exchange ?
- trouver du support commercial Linux ?
- utiliser du logiciel libre sur un environnement propriétaire ?

Echange de données On peut échanger des données avec des logiciels Microsoft Office : par exemple, on peut utiliser des applications libres et exporter en RTF. Citons notamment LyX (qui utilise le système L^AT_EX). Sous Linux existent aussi des applications commerciales comme StarOffice, Applixware et WordPerfect qui permettent de lire et d'écrire des documents aux formats Microsoft (Word, Excel, PowerPoint).

Une solution intégrée complète est bien sûr la suite OpenOffice.org. Elle permet d'assurer une très bonne compatibilité avec les composants de base de la suite bureautique Microsoft Office et propose des fonctions avancées supplémentaires.

Remplacer des serveurs On peut remplacer un serveur Microsoft Windows NT, Novell ou Apple MacOS dans le cas général par un serveur Linux, en particulier si celui-ci ne se charge que du stockage et du service des données. Pour les cas particuliers (applications supplémentaires : serveur de courrier électronique, base de donnée), c'est en général possible sans trop d'efforts (avec du tout libre, ou avec des logiciels propriétaires comme Oracle, Sybase, etc). Dans le cas de Novell, un émulateur pour les modules NLM existe (mars_nwe), et Novell fournit également ses logiciels propriétaires sous GNU/Linux.

Toute application utilisant **ODBC** ou **JDBC** devrait être portable sous Linux (du moins sa partie serveur) en conservant les scripts, programmes et interfaces utilisateurs précédentes.

Le cas de **Microsoft Exchange** est un peu particulier : ce serveur de courrier électronique SMTP, doublé d'un serveur POP/IMAP (et d'un protocole propriétaire supporté uniquement par Microsoft Outlook) est lié à un système de gestion de calendrier propriétaire. Pour migrer à une solution Linux (libre ou non) avec SMTP et POP/IMAP standard, il faut transférer les données stockées sur le serveur et remplacer la fonctionnalité calendrier. Dans un premier lieu, il suffit de retransférer les données via le serveur POP de **Microsoft Exchange**. Ensuite, il faut soit utiliser une application avec calendrier intégré (Star Office Linux ou Windows), ou une application séparée (p.ex. une des nombreuses applications PHP via WWW). Si l'on garde des clients Microsoft Windows et Microsoft Outlook, il existe au moins une implémentation du protocole propriétaire de calendrier pour Microsoft Outlook sous Linux, propriétaire.

Port de logiciels On peut porter un logiciel Microsoft Windows sous Linux de plusieurs façon. La moins coûteuse est de faire fonctionner ce logiciel sous l'émulateur libre **Wine**. Notons aussi que si l'application est bien conçue (séparation de l'interface graphique de la fonctionnalité), un port réel (natif) et la maintenance subséquentes des deux versions seront possibles à peu de frais. En particulier des bibliothèques comme Qt permettent d'écrire des logiciels en interface graphique à la fois sous Microsoft Windows et Linux.

Compatibilité MS-DOS/Win16/Win32 L'émulateur libre Wine permet de tourner certains programmes Microsoft Windows. Les programmes MS-DOS peuvent être exécutés avec l'émulateur MS-DOS **dosemu** (sous le MS-DOS compatible libre **FreeDOS**). On peut aussi lancer un système Microsoft Windows complet (dont il faut alors une licence Microsoft) en même temps que Linux avec le logiciel propriétaire **VMware**. VMware permet par exemple de lancer simultanément plusieurs versions de Microsoft Windows (versions différentes si nécessaires) pour tester de nouveaux programmes. On peut aussi configurer la machine en multi-boot. Attention, pour le moment tout cela n'est disponible que pour Linux/ix86 (pas pour Alpha, MIPS, PowerPC ou SPARC).

Compatibilité UNIX Tous les logiciels UNIX disponibles en format source tournent sous Linux, même si parfois quelques efforts de portage sont nécessaires, en particulier pour de vieux logiciels. Les logiciels disponibles en format binaire seulement (exécutable) ne tourneront pas en général sous Linux : une bibliothèque de compatibilité pour SCO UNIX et Solaris/ix86 existe. Alpha Linux peut tourner les exécutable Digital UNIX compilés statiquement, sans licence Tru64 UNIX.

Support Linux commercial De nombreuses entreprises offrent du support technique pour Linux, même si pour le moment il s'agit plutôt du support technique pour le développement de logiciels ou l'utilisation de Linux à grande échelle. Presque tous les distributeurs commerciaux offrent du support technique pour les utilisateurs finaux, gratuit pendant le premier mois.

Libre sous environnement propriétaires On peut tourner certains logiciels libres en dehors de UNIX ou de Linux : p.ex. OpenOffice.org, Mozilla, GIMP, Apache, htdig, ssh, perl et d'autres logiciels ont des versions Win32. Cela peut être le premier pas vers une migration complète et vous permettra d'évaluer ces applications.

\$Id: logiciels_libres.tex,v 1.3 2007/01/25 16:41:46 schaefer Exp \$