

Etude prospective sur les logiciels libres et commerciaux

Pascal Rulfi

Janvier 2009

Table des Matières

1. Résumé général	2
2. Logiciels libres et commerciaux	3
2.1. Licences.....	3
2.2. Normes et interopérabilité	4
2.3. Modèle technique	6
2.4. Modèle produits.....	7
2.5. Modèle économique	8
2.6. L'indépendance du logiciel libre	9
2.7. Conditions d'activité pour un logiciel ouvert.....	10
2.8. La gratuité et les coûts	10
2.9. Cycle de vie	11
2.10. L'homogénéité des solutions	12
2.11. Formation	12
2.12. Stratégie.....	13
2.13. Réservoir applicatif.....	14
2.14. Concrètement.....	14
2.14.1. Les enjeux de la bureautique	14
2.14.2. Internet et les serveurs	15
3. Critères de décision	17
3.1. Typologie des environnements.....	17
3.1.1. Commentaire	18
3.2. Matrice de décision.....	19
3.2.1. Commentaire	20
4. Conclusion.....	22
Pascal Rulfi.....	23

1. Résumé général

Depuis quelques temps, le logiciel libre semble constituer une alternative crédible au logiciel commercial. Cette alternative concorde avec un sentiment général hostile aux grandes firmes éditrices de logiciels.

Polarisés par le contrôle des dépenses et mus par une attitude inamicale vis-à-vis des éditeurs commerciaux, certains politiques demandent que la voie du logiciel libre soit sérieusement examinée par les instances techniques des administrations.

De plus, dans un monde qui propose des modèles gratuits, payants ou payés par des tiers, il devient difficile d'objectiver les modèles économiques. Devant des repères instables, les montants exigés pour l'utilisation de logiciels semblent élevés alors que la contrepartie est intangible.

Au-delà des querelles partisans, ce document a l'ambition d'aborder de façon neutre les enjeux du choix de produits logiciels.

Je souhaite m'affranchir de la part émotionnelle qui entoure ce débat et éviter les démarches simplistes telles que je les entends souvent.

Le choix d'une plate-forme logicielle est une décision stratégique qu'il convient d'aborder avec prudence et responsabilité. Il ne s'agit pas de choisir des solutions qui « semblent » bonnes mais des solutions qui préservent au mieux les intérêts de ceux qui consentent des investissements.

A défaut de constituer une étude comparative exhaustive, ce document traverse quelques sujets liés aux logiciels libres et commerciaux qui me semblent intéressants.

2. Logiciels libres et commerciaux

Je remarque que l'hostilité envers les logiciels commerciaux est plus particulièrement orientée vers Microsoft. Cette entreprise très exposée catalyse tous les mécontentements et toutes les frustrations des usagers de l'informatique.

Ainsi, l'ex président Bill Gates représenterait à lui seul l'image d'un chef d'entreprise vorace symbole d'un capitalisme agressif et impérialiste, ce qui laisse supposer que l'argumentaire des opposants est souvent des plus irrationnels.

A ce titre, je n'entends jamais la presse généraliste parler négativement de Larry Ellison d'Oracle ou de Larry Brain de Google. Pas plus que je ne lis des commentaires négatifs pour les sommes importantes exigées pour exploiter les logiciels tels Oracle, VMware, les antivirus et toutes autres solutions commerciales du marché.

De ce point de vue, j'observe clairement une attitude émotionnelle qui ne peut en aucun cas et à elle seule discriminer une solution sans autre analyse.

De plus, le grand public oppose le logiciel « gratuit » et le logiciel « payant » au couple Linux/OpenOffice contre Windows/MS-Office. Cette vue est fortement limitative et ne tient pas compte des contingences d'exploitation d'un système dédié à un usage professionnel.

2.1. Licences

Le **logiciel « propriétaire »**¹ ou « privé » repose sur un contrat qui stipule que le vendeur ne cède qu'un droit d'utilisation du logiciel.

Le contrat de licence utilisateur final (CLUF) repose sur le droit d'auteur. Le logiciel reste propriété de l'éditeur qui n'en cède que le droit d'utilisation.

En général, l'accès au code source du logiciel reste protégé. Cependant, l'éventuelle mise à disposition du code ne rend pas le logiciel libre pour autant.

Ce sont ces dispositions contractuelles qui sont connues des utilisateurs de logiciels d'éditeurs tels que Microsoft, Oracle ou Symantec et qui régissent le droit d'utilisation.

Le logiciel « libre », « ouvert », « gratuit » est surtout identifié comme du logiciel pour lequel on ne paie pas de licence, en dehors de toutes autres considérations.

La notion de logiciel libre² (free software) ne doit pas être confondue avec celle de logiciel gratuit (Freeware), ni de partagiciel (Shareware). De la même manière, le logiciel libre n'est pas équivalent au logiciel ouvert (OpenSource) car des différences contractuelles apparaissent alors que techniquement ces notions sont comparables.

Le *freeware* est un logiciel propriétaire qui est distribué gratuitement mais dont le code source reste protégé.

Le *shareware* est un logiciel propriétaire que l'on peut se procurer gratuitement mais qui devient payant si on souhaite l'utiliser.

¹ http://fr.wikipedia.org/wiki/Logiciel_propriétaire

² http://fr.wikipedia.org/wiki/Logiciel_libre

Le *free software*, l'*open source* et le *copyleft* (GNU) représentent des logiciels dont le code source est publié et libre d'usage, cela signifie que le code est modifiable et utilisable gratuitement.

Les différences entre ces différents modèles portent sur la formulation du droit de licence.

Le présent document n'a pas pour but d'aborder les aspects légaux relatifs aux différents modèles de contrat du logiciel ouvert. Les personnes intéressées se référeront aux articles mentionnés en bas de page qui renvoient aux licences GNU, BSD et Opensource³.

Au-delà des querelles de licences, on retiendra du logiciel libre que le code est publié sans garantie.

Il est également intéressant de relever qu'il n'est pas permis de privatiser un logiciel basé sur du code sous licence GNU. Toutes les contributions doivent rester ouvertes, ce qui peut être non souhaité dans certains cas.

Par conséquent, à l'heure du choix, il est opportun de contrôler les conséquences légales entourant le logiciel ouvert.

Notons encore à titre d'exemple que Mac OS X⁴, le système d'exploitation du Mac de la firme Apple, repose sur un UNIX FreeBSD.

La licence FreeBSD permet d'intégrer le système d'exploitation dans un projet commercial.

En d'autres termes, Apple s'appuie sur un logiciel libre pour construire son propre système MacOS dont le code est fermé et la licence payante.

Dans ce cas, Apple n'aurait pas pu procéder de la même manière avec un UNIX sous licence GNU (tel Linux) car son propre développement aurait hérité de la licence GNU et de ce fait n'aurait pas pu être licencié.

2.2. Normes et interopérabilité

L'aspect des normes est encore un sujet pour lequel circulent des lieux communs qui ne sont pas tout à fait exacts. La rumeur populaire déclare :

- « Microsoft ne respecte pas les normes »
- « le logiciel libre respecte les normes »

Il faut tout d'abord s'entendre sur ce qu'on appelle des normes.

ISO est une instance internationale et indépendante qui a la compétence d'éditer des normes. Le plus souvent, ces normes s'appuient sur des groupes de travail auxquels les éditeurs sont conviés.

Ces normes partent d'une intention louable et semblent empreintes de bon sens. Malheureusement dans la réalité les normes ne sont pas suivies à la lettre. Je vais l'illustrer par deux exemples.

La norme SQL.

SQL est un langage qui permet d'interroger et de manipuler une base de données.

Ce langage a été défini en 1970 par Edgar Frank Codd, un informaticien britannique.

IBM a développé SQL en 1970 pour ses bases de données relationnelles.

L'ANSI (institut de normalisation américaine) a adopté SQL comme recommandation en 1986.

L'ISO a normé SQL en 1987.

La norme a été amendée à plusieurs reprises, en 1986, 1989, 1992, 1999, 2003 et 2008.

Dans les faits, lorsqu'elle a été établie, la norme a été respectée par les différents éditeurs de base de données. Cependant, les éditeurs ajoutaient un certain nombre de fonctions qui n'étaient pas incluses dans la

³ Licence GNU : <http://www.gnu.org/licenses/gpl.html>

Licence BSD : <http://www.freebsd.org/copyright/license.html>

Licence OpenSource : <http://www.opensource.org/docs/definition.php>

⁴ http://fr.wikipedia.org/wiki/Mac_OS_X

norme mais qui facilitaient le travail.

Pour ne prendre que l'exemple de RdB de DEC, ce qui faisait partie de la norme était clairement indiqué dans les manuels de référence, libre au développeur de respecter la norme.

D'autre part, entre la première version de SQL et la première norme, il se sera écoulé 16 ans... La préhistoire à l'échelle de l'évolution informatique.

Je tire deux enseignements de cet exemple :

- L'informatique évolue à un rythme beaucoup plus rapide que les organismes de normalisation.
- Les besoins effectifs du terrain sont plus larges que ceux que les fonctionnalités généralement proposée par la norme.

La norme X500

X500 désigne l'ensemble des normes informatiques sur les services de messagerie et d'annuaire. Il s'agit d'une norme édictée par l'UIT (Union Internationale des Télécommunications).

Cette norme exhaustive a été développée dans les années 80 et a finalement été adoptée par l'ISO.

Malheureusement, cette norme imposée par des instances provenant des télécommunications ne correspondait pas aux besoins de l'informatique car elle était trop lourde pour l'époque.

Dans les faits, ce groupe de normes n'a formellement pas été adopté par le marché.

Pour l'anecdote, cette période a vu la rapide diffusion du fax pour la transmission de messages écrits. Cette machine ne respecte absolument pas la norme X500 et est une hérésie du point de vue de l'efficacité technique (toujours très mal intégrée au système d'information). Cependant le marché a plébiscité le fax contre l'avis de l'UIT.

D'autre part, en ce qui concerne les annuaires, X500 a été considérée si complexe qu'elle a été supplantée par LDAP.

LDAP (Lightweight Directory Access Protocol) est un protocole qui ne norme pas l'annuaire mais simplement l'interrogation et la modification de ce dernier.

DAP s'inspire de X500 mais en enlève toutes les lourdeurs, LDAP est une version allégée de DAP.

La norme LDAP est issue d'un ensemble de « RFC » (Requests for comments) qu'émet l'IETF (Internet Engineering Task Force). L'IETF produit la plupart des standards de l'internet.

Il faut mentionner que l'IETF n'est pas une instance officielle, il s'agit d'un groupe informel, sans statut, sans membres, sans adhésion, mais dont la légitimité est suffisante pour qu'elle face force de loi dans le monde de l'Internet !

Peu de RFC sont des standards, mais tous les standards d'Internet sont des RFC !

Les enseignements que je tire de cet exemple sont :

- **Normer n'est pas suffisant, il faut avoir l'adhésion du marché.**
- Quand on parle de norme, quelles sont les instances fiables ? Cela dépend du contexte...

Par conséquent, en informatique, **le standard est souvent une norme de fait**. En d'autres termes, **le plus fort l'emporte**.

Cette situation est assez peu confortable car **les choix qui sortent du rang peuvent s'avérer aventureux voire très coûteux et non viables**.

En définitive, si la voie de la normalisation est la plus sage, la réalité du terrain montre que la norme de fait est celle dont on peut attendre qu'elle définisse la norme internationale.

2.3. Modèle technique

Logiciel commercial

Le fonctionnement de la production de logiciel chez un éditeur de logiciel commercial est compris de tous. L'éditeur dispose d'une équipe de développement qui produit et met au point un logiciel qui sera commercialisé de façon traditionnelle.

Le support, dont la correction de bugs, est effectué par l'éditeur qui seul dispose du code source du programme.

Le logiciel commercial repose sur une organisation centralisée. Les objectifs sont décidés en haut lieu selon un plan défini de façon centralisée. Comme dans toute entreprise cotée, l'objectif est de maximiser les profits afin de satisfaire les propriétaires de l'éditeur.

L'éditeur commercial vit de la vente de ses produits, par conséquent il est condamné à créer de la nouveauté et se doit de proposer de nouvelles versions génératrices de chiffre d'affaire.

L'équipe de développement doit alors arbitrer entre plusieurs buts contradictoires : la fiabilité, les délais de mise sur le marché et les nouvelles fonctionnalités.

Force est de constater que cet arbitrage n'est pas toujours des plus heureux et peut déboucher sur la commercialisation de produits immatures.

Logiciel ouvert

Pour sa part, le logiciel ouvert part du principe que le partage de la connaissance accélère l'évolution. Tous peuvent contribuer à l'évolution du produit et le produit se diffuse d'autant plus vite qu'il n'y a pas de freins à la diffusion.

Nos acquis étant ce qu'ils sont, il est parfois difficile d'appréhender ce modèle de fonctionnement. Transposé à la fabrication d'un pont, cela signifierait que collectivement il y aurait la conscience de la nécessité de disposer d'un pont à un endroit précis et que chacun arriverait avec une brique, un concept ou du mortier et construirait, individuellement, une partie de ce pont alors que d'autres évalueraient la bonne tenue du pont. Dans le cas d'espèce, je pense que le modèle ne fonctionnerait pas et que le pont s'effondrerait. Et pourtant, dans le cas du logiciel, cela fonctionne !

Ce modèle, qui peut être perçu comme anarchique, représente la défense du bien commun dans une ultra-démocratie logicielle.

Les versions successives d'un logiciel ouvert ne sont pas imposées par le département marketing et on peut imaginer que ce type de fonctionnement ne fait pas la course à la fonctionnalité inutile.

En revanche, en cas de problème, il faut retrousser les manches et plonger dans le problème. Si les logiciels phares (Linux, Firefox, Apache) sont au moins aussi fiables que leurs homologues commerciaux, ce n'est malheureusement pas le cas des contributions marginales qui demandent souvent pas mal de temps pour les faire fonctionner correctement.

Notons pour l'anecdote que le logiciel libre est antérieur au logiciel commercial, des bibliothèques telles matlab ont été écrites en partage ; tels Monsieur Jourdain, les scientifiques faisaient du logiciel libre sans le savoir.

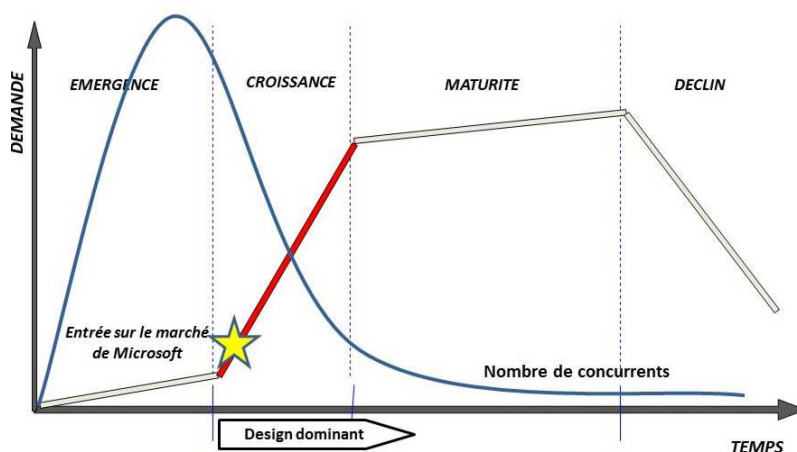
2.4. Modèle produits

Logiciel commercial

Rappelons que l'objectif d'un éditeur commercial est de satisfaire des actionnaires. Il s'agit pour eux de sortir des produits générateurs de marge et promis au succès.

Microsoft est parfois perçu comme une entreprise peu novatrice. C'est partiellement inexact car dans le cas d'espèce c'est le plus gros budget R&D du secteur. Cependant, il est vrai que Microsoft s'intéresse à un produit lorsque le marché est prêt à générer de la croissance.

Le cycle de vie d'un produit est décrit comme suit :



Microsoft entre sur le marché lorsqu'un potentiel de croissance a été identifié. Dans ce cas, le plus simple pour Microsoft est d'acquérir une entreprise active dans le secteur prometteur visé. Ce fut notamment le cas pour SQLserver, FrontPage ou Commerce Server.

Le bénéfice pour l'utilisateur de ce mode de fonctionnement est que la stratégie informatique est toute trouvée, il suffit de monter dans le train conduit par l'éditeur pour avoir une vue raisonnable de ce que peut être l'avenir.

Nous pouvons admettre qu'il s'agit d'un oreiller de paresse qui ne doit pas exclure la vigilance.

D'autre part, l'évolution d'un produit commercial est limitée par la force de frappe que l'éditeur est capable de mettre en œuvre. En d'autres termes il est nécessaire pour l'éditeur de disposer de vaches à lait capables de financer les développements prometteurs⁵.

Un éditeur commercial disposant d'un large portefeuille de solutions a la plus grande chance de durer dans le temps et la plus forte capacité théorique d'innovation.

C'est ainsi que fonctionnent des éditeurs tels Microsoft (par acquisition et développement de solutions prometteuses) ou CA et Symantec (par acquisition d'entreprises leader).

Logiciel ouvert

Le logiciel ouvert fonctionne de manière beaucoup plus empirique. Une fonctionnalité semble nécessaire (p.ex. un CMS), une petite communauté se met en place, développe un embryon de solution et la publie.

Toute la difficulté consiste à réunir une communauté suffisante pour faire croître la solution. Ce modèle est beaucoup moins déterministe que dans le cas d'une stratégie commerciale.

⁵ Cf modèle de portefeuille BCG (Boston Consulting Group)

Un projet de logiciel ouvert va chercher de faire « vibrer » le marcher et tenter de trouver une « résonance ». La lecture stratégique en est beaucoup plus difficile.

Un produit qui ne trouve pas sa communauté meurt de sa belle mort.

D'autre part, lorsqu'on devient contributeur, il devient difficile d'estimer la charge de la contribution, donc son coût. Une entreprise dont la production de logiciel ne constitue pas le cœur d'activité aura de la peine à juger ce que coûte réellement l'utilisation du logiciel ouvert et en quoi son équipe informatique apporte de la valeur à l'entreprise plutôt qu'à la communauté.

2.5. Modèle économique

Je n'épilogue pas sur le modèle économique du logiciel propriétaire qui est compris de tous ; la licence payée est la source du revenu de l'entreprise tel qu'on le connaît dans l'économie traditionnelle (paiement direct d'un bien).

Le modèle économique du logiciel libre est un peu plus flou. Au-delà de la vision un peu romantique du grand partage dans une vision de développement collectiviste (informaticiens de tous les pays unissez-vous...), il n'en reste pas moins que chacun perçoit un salaire d'une manière ou d'une autre.

La gratuité n'existe pas !

Cela mérite d'examiner d'où provient l'argent permettant le développement des logiciels libres.

La vision romanesque voulant que les informaticiens travaillent la nuit et participent à des projets collectifs pour le plaisir de voir leur nom associé à un logiciel doit être tempérée.

Si les bases d'un logiciel libres sont gratuitement accessibles, on constate que les auteurs de logiciels sont employés non pas à écrire du logiciel vraiment libre mais souvent à faire du sur mesure payant pour des commanditaires.

Il s'agit en fait d'un déplacement des charges d'autant que la part de l'investissement en logiciel d'infrastructure est faible vis-à-vis des coûts des développements des solutions métiers.

Avant d'effectuer un choix stratégique, il peut être intéressant d'examiner le modèle économique de la solution retenue afin d'évaluer sa viabilité dans la durée.

Quelles sont les sources de financement, quand bien même celles-ci seraient mutualisées ?

Qui arbitre l'évolution la solution et avec quels moyens ?

Examinons Ubuntu, éditeur d'une distribution entièrement gratuite du système d'exploitation Linux.

Cette distribution est soutenue par la société Canonical, entreprise fondée par Mark Shuttleworth, fondateur de Thawte, entreprise vendue en 1999 à Verisign pour 575 millions de US\$.

A ce jour (début 2009), Canonical n'est pas une entreprise rentable. Cela signifie que son propriétaire injecte les fonds nécessaires à son développement et a récemment affirmé être prêt à le faire encore 3 ou 4 ans. Par conséquent, à ce jour, l'avenir à 10 ans d'Ubuntu est incertain.

En revanche, RedHat qui édite également une distribution Linux est rentable. Cependant, si les logiciels restent ouverts, on ne parle plus de gratuité. En effet, la stratégie de RedHat est concentrée autour des serveurs qui deviennent payant au travers d'une souscription de support obligatoire.

RedHat a abandonné la distribution « RedHat Linux » en 2003. RedHat propose et soutient Fedora, la version grand public du système d'exploitation Linux.

D'autre part, on peut imaginer qu'un système d'exploitation pour téléphone mobile reposant sur Linux soit intéressant. En effet, la source de revenu provient de la vente du téléphone lui-même et non pas de son système d'exploitation.

Sans peindre le diable sur la muraille, au-delà de l'enthousiasme que peut provoquer le logiciel libre, il convient d'être prudent quant à l'avenir d'une solution.

Ce n'est pas parce qu'elle existe qu'elle a de l'avenir.

2.6. L'indépendance du logiciel libre

Le logiciel libre est associé à une forme d'indépendance et de liberté qui l'affranchissent du joug de quiconque.

La réalité montre que la frontière n'est pas aussi distincte entre le monde du logiciel libre et les entreprises commerciales.

Examinons quelques exemples :

OpenOffice :

Est une suite bureautique concurrente de Microsoft Office bien connue des usagers.

Depuis 2000, cette suite appartient à l'entreprise informatique SUN Microsystems. A ce jour, OpenOffice est toujours proposé sous une version libre.

MySQL :

Est une base de données libre leader dans son segment.

Le 16 janvier 2008, cette base de données devient propriété de SUN Microsystems⁶ (acheté 1 milliard de dollars). A ce jour, MySQL est toujours proposé sous une version libre.

Xen :

Est un hyperviseur de machine virtuelle libre concurrent de solutions telles VMware.

Août 2007, XenSource est racheté par Citrix pour un montant avoisinant les 500 millions de dollars.

Aujourd'hui, Xen est un produit intégré dans la gamme Citrix et commercialisé comme tel.

Il est légitime de se demander pourquoi des entreprises sont prêtes à déboursier des montants colossaux pour acquérir des produits « open » qui par nature génèrent de faibles revenus.

Dans le cas de Citrix, l'acquisition de **XenSource** est une opération stratégique qui permet à Citrix de devenir un acteur offrant tous les produits liés à la virtualisation, notamment du côté des serveurs où VMware est le leader.

On peut s'interroger sur le cas de **MySQL**. En effet, en 2007 le chiffre d'affaire estimé de MySQL était de 70 millions de dollars, le ticket d'entrée à 1 milliard de dollars payé par SUN pour rentrer dans le marché des bases de données semble exorbitant.

Officiellement il s'agit de créer un écosystème (JAVA, MySQL, Solaris, Apache) de même nature que celui qui a si bien réussi à Microsoft.

Il n'en reste pas moins que l'actionnaire de SUN Microsystems recherche avant tout la rentabilité et que ce type d'achat devra nécessairement générer du cash ; la question du modèle économique se pose.

A ce titre, on constate que la version « MySQL Community Server » est gratuite alors que la version entreprise est délivrée en quatre versions payantes.

Finalement, le modèle est le même que celui de Microsoft avec SQLserver, excepté que le code source de Microsoft n'est pas accessible.

Le cas d'**OpenOffice** est légèrement différent, en 1999 SUN acquiert à la société allemande Star Division le produit StarOffice.

StarOffice sera ouvert l'année d'après.

A cette époque les revenus de SUN proviennent principalement du hardware alors que les revenus de Microsoft dépendent du software et plus particulièrement du couple Windows/Office.

Il n'y a pas qu'un pas pour imaginer que cette opération avait pour but de couper l'herbe sous les pieds de Microsoft en donnant gratuitement un produit qui constituait la principale source de revenu de Microsoft.

⁶ Note actualisée : Avril 2009, Oracle Corporation annonce le rachat de SUN Microsystems, cet achat est avalisé en 2010 par la Commission Européenne. Oracle devient propriétaire de MySQL et d'OpenOffice. Une même société possède les deux produits concurrents que sont Oracle Database et MySQL, l'un est gratuit....

En tout état de cause, le monde du logiciel libre n'est assurément pas un représentant du bien contre le mal. Il convient d'examiner avec un peu plus d'attention les motifs de chacun et les modèles économiques qui prévalent afin de s'assurer de la pérennité et la stabilité des investissements.

2.7. Conditions d'activité pour un logiciel ouvert.

[Sourceforge](#) et [Freshmeat](#) créditent plusieurs dizaines de milliers d'applications libres. Les seules distributions standards ([Ubuntu](#), [Debian](#), [Red Hat](#)) comptent un millier de projets chacun.

Tous ces projets ont-ils un succès garanti ? Certainement pas !

Le code source d'un logiciel ouvert est publié et est en libre accès. Il est modifiable par chacun, la somme des contributions de la communauté permettra de faire évoluer le logiciel.

Encore faut-il qu'il y ait une communauté et qu'elle soit suffisamment importante en qualité et en quantité. Pour cela, il est nécessaire que le logiciel proposé suscite un intérêt suffisant pour que la communauté soit dynamique.

En d'autres termes, un logiciel ouvert peut mourir de sa belle mort faute de contributions. Pour s'en convaincre, il suffit de visiter des sites de partage de projets⁷ pour constater que tous les logiciels ne jouissent pas de la même activité.

D'autre part, la même fonctionnalité peut faire l'objet de plusieurs projets. Si des projets tels OpenOffice (suite bureautique) ou Apache (serveur web) trouvent un consensus et sont bien diffusés, le choix peut s'avérer plus délicat pour d'autres produits.

Par exemple, le choix d'un CMS (Gestionnaire de contenu web) peut être difficile. L'offre est abondante et la qualité pas toujours facile à évaluer ; Typo3, Joomla, SPIP, Drupal et bien d'autres sont autant de CMS intéressants⁸.

Lequel choisir ? Cette question est délicate car au-delà de la fonctionnalité, il faudra évaluer la maturité des produits, la facilité d'implémentation ainsi que le dynamisme de la communauté. Cela demande du temps et des compétences internes importantes.

En ligne de mire, c'est toute la question de la pérennité de la plate-forme qui se trouve ainsi posée. Question qui se pose également pour un logiciel commercial.

2.8. La gratuité et les coûts

Le logiciel libre est gratuit... Certes, mais à quel prix ?

L'idée n'est pas de détailler les coûts de possession de telle ou telle architecture car de nombreuses études ont été menées sur le sujet et arrivent à des conclusions qui dépendent du commanditaire.

En tout état de cause, la différence de prix est insuffisante pour faire basculer de manière nette le choix de la plate-forme.

Le logiciel ouvert ne pratique pas la licence, il est par conséquent gratuit.

La suite logicielle de Microsoft est payante et peut s'avérer coûteuse pour une PME. Cependant, les administrations publiques négocient des contrats avec les éditeurs dont les conditions financières sont très favorables aux bénéficiaires.

⁷ <http://sourceforge.net>

⁸ <http://www.cmsreview.com/>

Ces acquisitions de licences peuvent paraître coûteuses aux instances exécutives qui doivent inscrire ces investissements au budget. Sans pour autant minimiser l'importance des montants engagés, il convient de mettre ces montants en perspective.

Par exemple en amortissant les investissements sur 3 ans et en les rapportant par collaborateur, on s'aperçoit vite qu'en regard d'un salaire, le montant de l'investissement pour l'outil de production n'est pas un critère déterminant pour faire un choix.

La migration vers une autre plate-forme est une aventure coûteuse. Au-delà de l'économie de licences, il faut notamment investir dans :

- L'ingénierie de migration,
- L'adaptation, voire la réécriture des programmes
- La formation des collaborateurs

Le tout pour un bénéfice fonctionnel nul et une prise de risque considérable.

Dans ces conditions, il n'est pas étonnant de constater que très rares sont ceux qui ont effectivement migré toute leur infrastructure vers le logiciel libre.

Et ceux qui tentent l'aventure n'enregistrent pas des succès déterminants.

2.9. Cycle de vie

Tout un chacun vit l'expérience d'une informatique qui évolue extrêmement rapidement. Les versions majeures des différents logiciels se succèdent tous les 18 mois créant l'illusion d'une mouvance continue qui ferait abstraction des acquis.

Notons que cette mouvance est le fait d'applications standards largement diffusées.

De mon expérience, j'observe que les cycles technologique ont une période d'environ 7 ans.

Des gros ordinateurs centraux aux minis, puis au PC, au réseau de PC, au multimédia, à Internet et la convergence, tous les 7 ans une nouveauté marquante apparaît et fait vaciller les entreprises en place.

Il suffit de penser à des firmes telles IBM, Digital Equipment, Compaq, Data General, Word Perfect, Lotus 1-2-3 ; ce sont autant d'entreprises qui ont connu la gloire et qui maintenant ont été fortement redimensionnées quand elles n'ont simplement pas disparues.

Alors que des géants comme Google étaient totalement inconnus il y a seulement 10 ans.

Lorsqu'une entreprise ou une collectivité consent à développer une solution informatique pour un besoin spécifique, elle engage des moyens importants et souvent stratégiques.

Si on a parfois l'illusion de développer une solution pour un besoin temporaire, la durée de vie réelle d'un logiciel métier est très souvent supérieure à 10 ans.

Ce sont ces solutions métiers qui devraient constituer la principale préoccupation des stratégies de l'entreprise, non pas les discussions de comptoir sur la plate-forme idéale au temps « T ». Par ailleurs, c'est souvent le logiciel métier qui détermine la plate-forme qui hébergera l'application, pas le contraire.

Cette petite digression historique pour montrer qu'une application métier représente un investissement conséquent qu'il s'agit de préserver plusieurs lustres. Le véritable enjeu consiste à s'appuyer des partenaires fiables dans la durée afin de minimiser les mauvaises surprises.

Dans le monde de l'informatique où les évolutions sont constantes et les révolutions toujours possibles, il est difficile de déterminer ce qu'est un fournisseur fiable.

Au-delà des envies et des affinités techniques des informaticiens, il convient d'examiner si le fournisseur d'une plate-forme dispose d'une part de marché suffisante pour constituer un écosystème viable ainsi que des moyens financiers pour faire vivre et évoluer sa plate-forme.

Sous cet angle de vue, les fournisseurs alternatifs ne sont pas légion, et leurs propositions ne répondent pas toujours de manière convaincante aux critères liés à la durée.

2.10. L'homogénéité des solutions

La technique informatique n'est manifestement pas simple quand bien même les services marketing vantent une simplicité de façade.

La société Apple fournit une vue tangible de ce que peuvent être des produits simples et homogènes. iPhone, Mac et iPod sont des produits qui bénéficient d'interfaces tournées vers l'utilisateur et homogènes entre produits.

La navigation entre ces produits pourtant différents s'effectue de la même manière, une fois l'esprit « Apple » acquis. Cet aspect constitue à mon avis le facteur clé de succès actuel de la société à la pomme. La stratégie d'Apple est manifestement tournée vers les utilisateurs finaux qui plébiscitent ses solutions (avec raison).

Microsoft joue sur une homogénéité technique, chaque famille de produit est accompagnée d'une boîte à outils assez homogène dont la philosophie de fonctionnement est proche d'une application à l'autre. L'environnement de développement .NET peut être vu comme un élément fédérateur des solutions proposées par Microsoft.

Microsoft a une stratégie tournée vers l'entreprise, une majorité de briques informatiques qui intéressent l'entreprise font partie du portefeuille produit de Microsoft.

On remarque par ailleurs que les produits grand public de Microsoft (baladeur, téléphone) ne connaissent pas un succès aussi éclatant.

Le logiciel ouvert propose énormément de solutions couvrant toutes sortes de besoins. Par nature, les projets ne sont pas fédérés avec pour conséquence des solutions peu homogènes. Ce constat est également observable au sein du même projet où les pièces rapportées (plug-in) portent la patte de leur concepteur. Cela signifie que la charge pour l'apprentissage de chaque fonctionnalité est plus élevée car moins intuitive et par une approche moins mécaniste.

Par nature, le logiciel ouvert demande des compétences spécialisées et des personnes plus agiles pour maîtriser les différentes approches de chaque solution.

Le logiciel libre s'adresse plutôt à une communauté de spécialistes tels les scientifiques ou les informaticiens.

2.11. Formation

Une majorité d'éditeurs de logiciels commerciaux ont délégué à des entreprises tierces le support de leurs produits.

Ce mode de fonctionnement est financièrement intéressant pour ces entreprises qui se concentrent dans le développement et trouvent un formidable bras de levier pour commercialiser et implémenter leur solution.

Pour garantir une qualité de service convenable, ces firmes ont mis en place des cycles de formations certificatives.

Décriées par les informaticiens, ces certifications ne constituent peut être pas la panacée mais sont une base réaliste pour qualifier les compétences des prestataires (internes ou externes).

Cette approche systématique permet de trouver un réservoir de prestataires qui rend le support plus fluide en réduisant les dépendances à des ultras spécialistes.

Par son approche non centralisée, le logiciel ouvert n'est pas fédéré et laisse une grande liberté à chacun d'agir comme bon lui semble. On trouve des applications géniales mais qui sont lâchées dans la nature sans autre support.

Les utilisateurs devront s'initier à l'application et se forger une compétence dans un environnement alternatif. Livres lorsqu'ils existent, forums et communauté forment une autre façon de travailler moins déterministe que les cursus prévus par les éditeurs.

Le logiciel libre est séduisant pour les informaticiens qui trouvent une façon d'exprimer leur talent et prouver leur qualité en proposant des solutions alternatives intéressantes.

En revanche, le chef d'entreprise prend le risque de créer une dépendance accrue envers des personnes détentrices de compétences pour l'instant rares.

Notons que les universités sont des creusets de compétences pour le logiciel libre.

Je peux imaginer que la génération des informaticiens qui sortent aujourd'hui des universités accéderont demain à des postes décisionnels, ils auront ainsi la potentialité de prendre des décisions en faveur du logiciel libre. J'estime que cette hypothèse peut devenir une réalité dans une échéance de dix ans.

2.12. Stratégie

Le long terme

Plus de 90% des ordinateurs fonctionnent actuellement sous Windows. On peut regretter le manque de diversité et se poser beaucoup de questions sur un marché qui choisit une solution payante alors qu'il existe de très bonnes solutions gratuites.

Malgré les changements de versions réguliers de Windows et les désagréments que les migrations entraînent, je remarque une relative stabilité technologique depuis une quinzaine d'années.

Le bénéfice pour les clients de cette relative stabilité est un investissement qui a été relativement protégé pendant cette période.

Même si des migrations et évolutions ont été nécessaires, il y a toujours une voie de migration qui permet de préserver l'investissement.

Les décisions novatrices peuvent s'avérer aventureuses voire catastrophiques lorsqu'elles mènent vers un cul de sac.

Il me semble imprudent d'aller contre le marché lorsqu'on ne dispose pas d'une taille critique suffisante. Les grandes entreprises à la tête d'un parc important et qui exploitent des applications d'entreprises limitées peuvent sereinement envisager des solutions alternatives telles que le logiciel ouvert car outre la masse critique, elles sont technologiquement autoporteuses.

Il appartient au décideur d'examiner sa propre configuration et d'évaluer l'opportunité des choix technologiques.

L'indépendance

En matière de dépendance technologiques, il faut remarquer une forte dépendance envers les Etats Unis, que ce soit pour le matériel ou pour les logiciels.

L'Europe prend sa revanche pour les secteurs des ERP (p.ex. SAP) ou CAD/CAM (p.ex. Catia de Dassault) L'Europe pourrait trouver un intérêt stratégique à acquérir une indépendance en matière de technologies informatiques mais ce projet serait de dimension européenne.

Cependant, il est légitime de se demander s'il est du ressort de l'Etat de développer du logiciel. A plus forte raison il appartient au citoyen de décider s'il est opportun d'entretenir une communauté de développeurs pour contribuer au logiciel libre ; pour quels objectifs et pour quels bénéfices.

En tout état de cause, ce genre de question doit être posé à une échelle nationale, une petite structure doit se méfier des opérations potentiellement aventureuses.

L'évolution

Les entreprises commerciales éditrices de logiciels doivent satisfaire des actionnaires en présentant des comptes bénéficiaires.

Par conséquent, ces entreprises doivent produire des logiciels rentables donc appréciés par les clients. Je relève de cette évidence un effet de bord intéressant : lorsqu'un éditeur lance un produit, il y a une bonne probabilité que cela corresponde à un besoin effectif du marché. En d'autres termes, on peut ainsi extrapoler à peu de frais les tendances fortes.

Pour de grands éditeurs, il est nécessaire de trouver des relais de croissance car il n'est pas concevable de vivre sur la seule rente de situation que procure un logiciel historique.

Concrètement, Microsoft n'a pas pu générer de la croissance pendant 20 ans uniquement sur le couple Windows/Office.

SQL Server pour les bases de données, Exchange pour la messagerie, Dynamics pour les ERP et maintenant SharePoint pour le travail collaboratif sont autant d'offres qui ont relancé au moment opportun les ventes de l'éditeur.

Par extrapolation, il suffirait de suivre ce type d'éditeur pour disposer d'une stratégie informatique simple et qui minimise le risque.

2.13. Réservoir applicatif

Le marché profite d'un acteur hégémonique. Tout éditeur qui développe une solution va prioritairement l'adresser là où le marché est le plus grand.

L'environnement au bénéfice de la plus grande part de marché se trouve renforcé dans sa position. Ce qui peut sembler très frustrant est d'une logique économique implacable.

Accessoirement, nous avons vu que c'est le même type de logique qui anime le logiciel ouvert car il faut une communauté forte et nombreuse pour qu'une application soit entretenue.

Pour un même type de fonction (p.ex. CMS) c'est la solution forte de la plus grande communauté qui risque de survivre.

2.14. Concrètement

Sur quelques objets, je souhaite apporter des observations afin d'illustrer concrètement les enjeux.

2.14.1. Les enjeux de la bureautique

Attardons nous sur le sujet de la bureautique.

La question demeure, pourquoi payer une suite bureautique alors qu'une version gratuite est disponible ?

La question est parfaitement pertinente car les fonctionnalités de Microsoft Office et d'OpenOffice sont plus ou moins équivalentes.

Une réponse pourrait être la compatibilité imparfaite des formats de documents. En effet, un document complexe n'est jamais parfaitement lu d'un outil à l'autre et à plus forte raison lorsque que les documents sont modifiés de part et d'autre.

Autant se plier à la majorité des utilisateurs qui, aujourd'hui, exploitent Microsoft Office.

D'autre part, une migration d'un fournisseur de bureautique à l'autre demande une migration des modèles et éventuellement des documents. Ce type de migration est par nature coûteux.

Il y a une raison stratégique pour laquelle il faut être attentif au choix du système de bureautique.

En effet, actuellement, OpenOffice et MS-Office ont tous deux des formats de documents qui séparent plus clairement le contenu du document de sa présentation.

Cette évolution peut paraître anodine mais, à mon avis, elle ouvre de nouvelles possibilités en matière de traitement documentaire.

Workflow, traitement automatique, travail collaboratif sont autant de fonctionnalités qui vont se développer à l'avenir.

Implémenter ce type de solution n'est pas seulement un enjeu technique mais engage des compétences organisationnelles et des talents de négociation pour modifier les processus en entreprise. Il ne s'agit plus de mettre en place des solutions techniques mais également de repenser la façon de travailler.

Par exemple : l'administration en ligne devrait permettre au citoyen de commander des documents ou des services par le web. Ces documents peuvent être partiellement gérés ou produits dans un flux d'information automatisé. Cette automatisation reposera sur des outils de workflow.

Par exemple : le partage de documents par plusieurs personnes prend du sens lorsqu'il est facile de retrouver des références. La simple gestion hiérarchique telle qu'elle est pratiquée dans un classement en répertoire est insuffisante. L'organisation et la gestion des documents demandent des outils de gestion documentaire ou des intranets sophistiqués.

Je crois que par la nature de leur activité, les administrations sont concernées par tout ce qui touche à l'optimisation de la gestion documentaire. En effet, ces outils seront des facteurs d'augmentation de la productivité et d'efficacité qui intéresseront les administrations au premier chef.

Comme on le voit, le choix de la plate-forme bureautique n'est pas anodin car les enjeux dépassent largement la simple rédaction de lettre ou la réalisation de tableaux simples. Outre l'outil bureautique, il convient d'analyser l'offre existante en périphérie des produits Office.

Les projets d'intégration documentaire sont complexes car ils exigent des compétences multiples. La simplicité des outils permettant d'implémenter ce genre de fonctionnalité et la garantie d'évolution dans le temps sont, à mon avis, des éléments à analyser lors du choix.

D'autre part, nous sommes probablement au début des processus d'automatisation de la production administrative et il convient de faire des choix de produits qui minimisent les impasses.

2.14.2. Internet et les serveurs

Le domaine du poste de travail ne semble pas être l'endroit privilégié par les entreprises pour introduire l'OpenSource. Pour toutes sortes de raisons, cette position est totalement justifiée. En revanche, le logiciel libre rencontre plus de succès pour la partie liée aux serveurs et leur périphérie ainsi que pour les services axés autour du web.

Un peu plus de 50% des entreprises européennes⁹ utilisent des logiciels Open Source. Environ 70% l'utilisent pour des domaines liés au développement et aux serveurs alors que seuls 20% l'utilisent pour les systèmes d'exploitation des postes de travail.

Il faut interpréter ce résultat car il n'informe pas sur la part de marché interne de l'Open Source. En d'autres termes il suffit d'exploiter 2 postes en Linux sur un parc de 5'000 machines sous Windows ou Unix pour prétendre avoir adopté des technologies Open Source.

Il apparaît que le logiciel Open Source n'est pas utilisé de façon marginale. En revanche, il est peu visible des utilisateurs finaux qui ont le sentiment d'être enfermés dans le carcan Windows/Office.

L'ouverture vers d'autres environnements que celui de Microsoft est notamment argumenté par la volonté de ne pas se laisser enfermer chez cet éditeur.

Cet argument est recevable car nous estimons que Microsoft a tendance à verrouiller ses utilisateurs.

L'utilisation d'une de ses solutions en amène une autre.

Par exemple, MS-Office peut amener l'environnement collaboratif SharePoint qui peut impliquer la base de données SQL Server. Le terrain sera ainsi occupé par des produits de la firme de Redmond.

⁹ JdN, enquête : Survey Interactive

En tout état de cause, le logiciel ouvert propose de très intéressantes solutions sectorielles qui méritent d'être analysées.

Lorsqu'on cherche un outil pour réaliser telle ou telle fonction, il n'est pas insensé de chercher du côté des logiciels ouverts. L'offre est pléthorique et cela permet de se familiariser avec ce type de technologie dont l'approche est différente que les voies toutes tracées des logiciels commerciaux.

3. Critères de décision

Au travers d'outils empiriques je vais tenter d'établir quelques critères d'aide à la décision.

Les critères et leur évaluation sont produits sur la base d'une appréciation personnelle qui repose sur mon expérience.

Je les émets dans un esprit le plus objectif possible et je souhaite écarter les aspects émotionnels de mon jugement. Il n'en reste pas moins qu'il s'agit d'une appréciation humaine avec toutes les libertés que cela suppose.

3.1. Typologie des environnements

Chaque environnement a ses forces et ses faiblesses qui en font ses qualités. Je vais tenter de définir quelques critères et noter sur une échelle de 1 à 5 (de mauvais à excellent) les plateformes logiciel commercial (Microsoft) et logiciel ouvert (Linux).

Par plate-forme « Linux », je comprends un ensemble cohérent constitué de logiciels ouverts qui permettent d'exploiter une infrastructure complète.

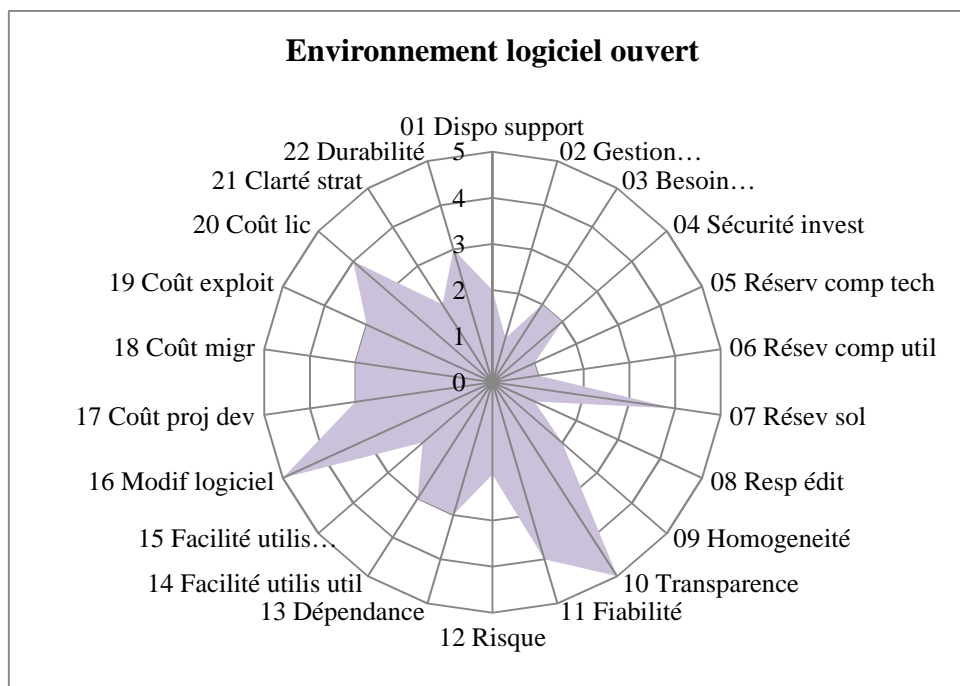
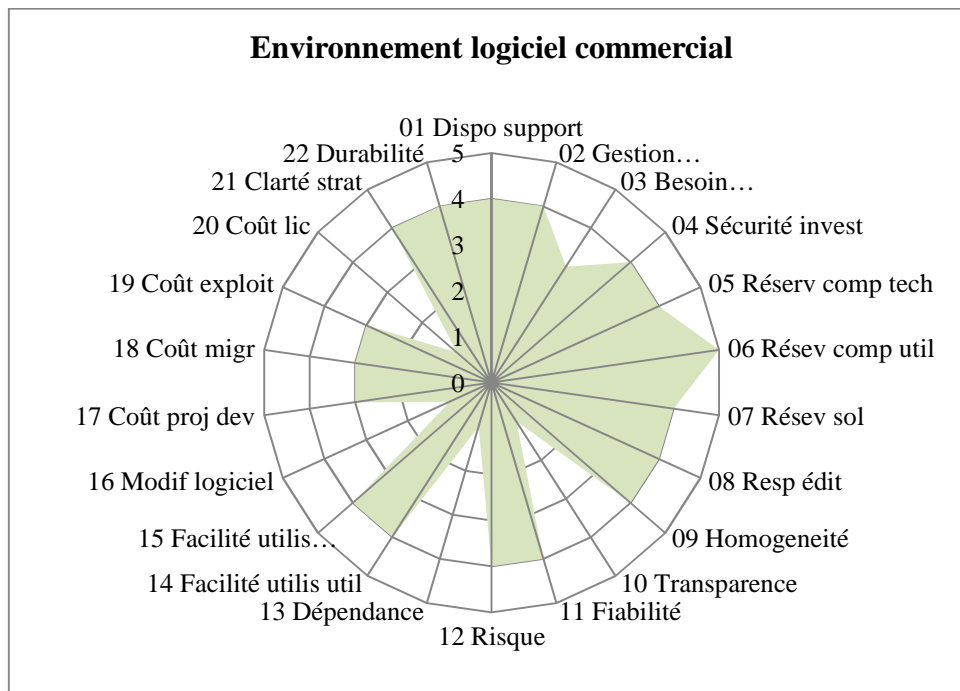
Par plate-forme « Microsoft », j'entends un ensemble de logiciels d'origine commerciale permettant d'exploiter une infrastructure complète et cohérente. A noter que je n'attends pas que tous les logiciels commerciaux proviennent de Microsoft.

Par mesure de simplicité, je ne mélange pas les deux modèles (p.ex OpenOffice sur Windows)

Je propose 22 critères qui me paraissent intéressants :

Critères	Microsoft	Linux
01 Disponibilité du support	4	2
02 Structure de gestion de la compétence	4	1
03 Nécessite des compétences élevées	3	2
04 Sécurité de l'investissement	4	2
05 Réservoir de compétences techniques	4	1
06 Réservoir de compétences utilisateur	5	1
07 Réservoir de solutions	4	4
08 Responsabilité de l'éditeur	4	1
09 Homogénéité des solutions	4	2
10 Transparence de l'éditeur	1	5
11 Fiabilité de l'environnement	4	4
12 Risque d'implémentation	4	2
13 Dépendance à un fournisseur (et un pays)	1	3
14 Facilité d'utilisation pour les utilisateurs	4	3
15 Facilité d'utilisation pour les équipes techniques	4	2
16 Possibilité de modification du logiciel	1	5
17 Coût d'un projet de développement	3	3
18 Coûts de migration	3	3
19 Coûts d'exploitation	3	3
20 Coûts de licences	1	4
21 Clarté de la stratégie de l'éditeur	4	2
22 Durabilité de la solution	4	3
<i>Moyennes</i>	3.32	2.64

Les images radar correspondantes au tableau illustrent les résultats.



3.1.1. Commentaire

Je rappelle que cette évaluation est le résultat d'une appréciation personnelle. En l'espèce, elle ne repose pas sur des études d'opportunité effectuées par des cabinets d'audit connus mais sur un seul avis basé sur ma modeste expérience.

Tout d'abord, les moyennes obtenues montrent qu'il n'y a pas une solution suffisamment mauvaise pour être clairement éliminée. De la même manière, aucune des deux solutions ne collectionne des notes excellentes.

En revanche, la solution commerciale dispose d'une moyenne plus élevée que le logiciel ouvert. Dans cette évaluation, le logiciel ouvert est juste au-dessus de la moyenne alors que le logiciel commercial reste globalement bon.

A la lumière de ces résultats, il paraît naturel que les milieux scientifiques et académiques se tournent vers le logiciel libre. La possibilité de modifier le logiciel, d'utiliser sans restriction toutes sortes d'applications (qui ne seront pas toujours exploitées), de s'affranchir d'un coût de licence et de pouvoir participer à une communauté correspond parfaitement aux attentes de ce type d'utilisateurs.

Les entreprises et les administrations qui exploitent l'outil informatique et qui attendent du rendement à un coût maîtrisé en minimisant le risque ont une démarche plus mécaniste de l'outil informatique. Les structures offertes par les éditeurs sont rassurantes et déterministes, ce qui est séduisant pour les clients de ces éditeurs.

Enfin, les coûts de licences sont souvent mis en avant par les détracteurs des solutions commerciales mais ils restent marginaux dans la totalité des coûts engendrés par l'informatique. Les coûts de licence ne représentent pas un critère déterminant.

3.2. Matrice de décision

Dans ce chapitre, je tente une approche par une classique matrice SWOT

Je ne m'occupe que de la plate-forme client (postes de travail) et traite trois cas.

Hypothèse 1 : conservation de la plate-forme Microsoft.

Force	Faiblesse
<ul style="list-style-type: none"> - Prise de risque limitée - Capital connaissances préservé - Capital expérience exploité - Capital logiciels préservé 	<ul style="list-style-type: none"> - Dépendance à un éditeur - Coûts des licences
Opportunité	Menace
<ul style="list-style-type: none"> - Centrer les ressources sur un apport de fonctionnalités - Chantiers métiers intéressants 	<ul style="list-style-type: none"> - Décisions politiques - Image dégradée, pression politique
Commentaire	
Ce cas de figure est bien entendu le plus raisonnable. Dans un environnement connu, les ressources peuvent/doivent être attribuées à l'apport de valeur pour les clients.	

Hypothèse 2 : migration à OpenOffice sur plate-forme Windows

Force	Faiblesse
<ul style="list-style-type: none"> - N'impose pas de changement d'interface comme Office 2007 	<ul style="list-style-type: none"> - Non compatibilité avec la majorité du marché. - Incertitudes quant au succès du projet
Opportunité	Menace
<ul style="list-style-type: none"> - Premier pas relativement indolore dans le logiciel libre - S'affranchir du « joug » de Microsoft. 	<ul style="list-style-type: none"> - Migration des documents existants incertaine - Intégration avec les serveurs de workflow et collaboration
Commentaire	
<p>OpenOffice pourrait constituer un premier pas dans le monde du logiciel libre. Dans le cas d'une administration publique, le coût de licence ne peut être porté au crédit de cette proposition car Microsoft propose une tarification très agressive qui inclut divers produits dont Office.</p> <p>Ce type de migration se voudrait être un « signal politique ». Je le trouve vain car il n'y a rien à gagner fonctionnellement ou financièrement. En revanche il existe un risque qui reste à évaluer.</p>	

Hypothèse 3 : migration sur une plate-forme open soft (Linux + OpenOffice +...)

Force	Faiblesse
<ul style="list-style-type: none"> - Projet fédérateur - S'affranchir de logiciels américains 	<ul style="list-style-type: none"> - Risque important - Projet aventureux - Manque de connaissances et de compétences - Capital logiciels métiers
Opportunité	Menace
<ul style="list-style-type: none"> - S'affranchir de l'environnement d'un éditeur hégémonique. - Renforcer les compétences des équipes informatiques. 	<ul style="list-style-type: none"> - Chantiers métiers - Dérapages des coûts
Commentaire	
<p>C'est un projet conséquent, coûteux, risqué et qui n'apporte rien aux utilisateurs. De plus, à ressources constantes, il va fortement diminuer les moyens attribués à la génération de valeur (applications métier, réduction des coûts)</p>	

3.2.1. Commentaire

Dans ces conditions, j'admets ne pas être de fervents défenseurs du logiciel libre, du moins pas dans l'environnement client.

Si dans les années 2003 à 2005 beaucoup de collectivités publiques ont annoncé en fanfare leur intention de migrer leur environnement vers le logiciel ouvert, force est de constater qu'à l'usage, les réalisations ont été beaucoup moins ambitieuses.

Pour prendre l'exemple de la mairie de Munich, elle avait fait parler d'elle en 2003 avec un ambitieux projet de migration. Problèmes de droit de licence, dépassement de budget, retards, difficultés techniques ont émaillé le projet.

Ainsi en septembre 2006, seuls une centaine de postes sur les 14'400 fonctionnaient sur du logiciel libre.

Les dernières nouvelles parlent d'une migration qui devrait s'achever en 2009. Dans tous les cas, la communication est actuellement nettement moins triomphante pour ne pas dire inexistante. Il est admis que des postes ne seront pas migrés pour des raisons de compatibilité avec certaines applications métier.

Le budget avoué serait de 35 millions d'Euros. Nous ne savons pas ce qui est réellement inclus dans ce budget car un certain nombre d'applications métier devront être converties ou réécrites ce qui impliquera des investissements autrement plus conséquents.

Tous ces investissements n'apporteront absolument aucun bénéfice à la population.

Au final, le bilan de cette opération peut être sévèrement jugé ; caprice d'ingénieur ou opération de marketing ? En tous les cas cela illustre ce que peut être un projet aventureux.

Il n'en reste pas moins qu'il existe des projets de migration réussis. Nous notons qu'il s'agit souvent de structures plus petites ou de structures dont les utilisateurs sont limités à la bureautique et une ou deux applications métier bien définies.

D'autre part, je n'ai pas émis d'avis sur la partie serveur de l'infrastructure. Il y a, à mon avis, d'intéressantes expériences à mener sur ce type d'infrastructure, en particulier en ce qui concerne la partie technique (surveillance, mesure, contrôle, etc) et les environnements web (serveurs, etc).

Le monde du web, repose sur des cycles de vie beaucoup plus courts que le logiciel de gestion, il est par conséquent plus facile d'introduire du logiciel libre et cas échéant modifier ou changer les outils. De plus l'évolution intrinsèque du web demande une plus grande agilité dans l'approche des services que l'on souhaite publier. Le logiciel ouvert devient alors une proposition adaptée à cet environnement.

4. Conclusion

L'informatique est un outil censé améliorer la productivité. Le retour sur investissement et l'apport de valeur pour le client devrait être le seul souci des stratèges et des décideurs lors de l'attribution des ressources.

L'informatique est un sujet qui suscite les passions et aiguise les querelles de clocher. Croyance et lieux communs sont souvent au centre des nombreuses discussions de café du commerce. Au demeurant, l'infrastructure est une affaire de spécialistes qu'il convient d'aborder avec toute la prudence et l'expérience requises car les investissements sont conséquents.

Il ne s'agit pas de condamner telle ou telle solution mais de tirer le meilleur pour le bénéfice des usagers. Ceci en évaluant sereinement l'ensemble des critères qui forment un choix responsable.

A mon avis, les chantiers qui méritent l'attention sont ceux qui visent à augmenter la productivité des entreprises et facilitent la vie, soit tout ce qui apporte de la valeur.

Pour les administrations ces enjeux impliquent un changement profond dans les processus qu'il s'agit de mécaniser et automatiser au travers de l'outil informatique.

Devant l'importance et la complexité de cette tâche, les discussions et remises en cause de la plate-forme me semble un débat d'arrière-garde.

Je ne minimise pas les changements technologiques qui pourront avoir lieu, l'hégémonie d'IBM a fait long feu, le tour viendra pour Microsoft. Cependant, il me semble prématuré de faire une remise en cause profonde de la plate-forme actuelle.

Notamment, une migration vers le logiciel ouvert de la plate-forme client me semble tout à fait aventureuse et contre-productive, du moins dans une échéance de cinq à dix ans.

En revanche, le logiciel ouvert est pertinent pour des outils techniques ou liés au web.

Cette présence aura peu de visibilité auprès des utilisateurs, au grand dam de certains. Cependant une telle orientation peut constituer des projets motivants pour le personnel technique et permet de démarrer une approche constructive et peu périlleuse vers le logiciel libre.



Pascal Rulfi

Le rédacteur du présent rapport bénéficie d'une formation d'ingénieur effectuée à l'Ecole d'Ingénieur de Genève ainsi qu'à l'Ecole Polytechnique Fédérale de Lausanne. De plus il est au bénéfice d'un Master in Business Administration réalisé à l'Université de Genève.

Pascal Rulfi a fondé et anime depuis plus de vingt ans la société Windsoft. Il est notamment en charge des mandats de consultance stratégique et organisationnelle liés aux technologies de l'information.

© Pascal Rulfi, 2009
<fin de document>