

Logiciels libres... Liberté ? Transparence ? Coopération ?

Dans le cadre des rencontres d'ADELI... autour d'un verre

Martine Otter

Lundi 3 avril 2006, ADELI organisait sa troisième « rencontre autour d'un verre », dans le cadre de la Villa Maillot. Ces rencontres rassemblent, pendant deux heures, adhérents et non-adhérents autour de thèmes liés aux systèmes d'information. Les deux premiers sujets retenus étaient ITIL, présenté par Thierry Chamfrault, puis « les processus », introduit par Didier Dussard. Le thème retenu cette fois était la problématique des logiciels libres. Le conférencier volontaire pour cet exercice oral était Benoît Sibaud, Président de l'APRIL, Association pour la Promotion et la Recherche en Informatique Libre. L'exercice consiste, rappelons-le, à présenter le sujet en une demi-heure, sans support, puis à mener le débat avec la salle, tout en prenant un verre...

Nous vous proposons ci-dessous un bref compte rendu de cette soirée et vous invitons à consulter régulièrement l'agenda d'ADELI pour venir participer avec nous aux prochaines soirées.

Le conférencier se présente

En tant qu'expert des Logiciels Libres et Président de l'APRIL, Benoît Sibaud intervient dans de nombreuses conférences et publications. Il est directeur des programmes des « Rencontres Mondiales du Logiciel Libre ». Il participe à des émissions de radio, comme « **01 Business** » sur **BFM**. Il était au Carrefour Numérique de la Cité des Sciences et de l'Industrie lors de l'édition 2005 de « **Libre En Fête** »...

Le conférencier nous parle des logiciels libres

Définitions

Quelques définitions s'imposaient ; Benoît Sibaud nous présenta donc en introduction les quatre libertés associées à la notion de logiciel libre :

- la liberté d'utiliser et d'exécuter le logiciel pour quelque usage que ce soit ;
- la liberté d'étudier le fonctionnement du programme, et de l'adapter à ses besoins propres (libre et open source sont des termes synonymes) ;
- la liberté d'en redistribuer des copies ;
- la liberté de modifier le programme, et de rendre publiques les améliorations apportées, afin que la communauté tout entière en bénéficie.

Un logiciel libre peut ainsi être utilisé, copié, étudié, modifié et redistribué sans restriction. Il rappela également qu'il ne fallait pas confondre « libre » et « gratuit » : les freeware, gratuits, ne sont pas fournis avec les sources et ne sont donc pas modifiables. De même les shareware ou partagiciels restent fermés et leur auteur demande souvent une contribution aux utilisateurs.

Il ajouta également qu'il existe des logiciels libres commerciaux et d'autres non commerciaux, comme il

existe des logiciels propriétaires commerciaux et d'autres non commerciaux.

Les brevets logiciels

La menace la plus importante vis-à-vis des logiciels libres est celle que fait peser la notion controversée de brevetabilité des logiciels.

La législation en matière de brevetabilité des logiciels varie en effet suivant les pays : les USA et le Japon accordent des brevets aux logiciels, alors que la législation européenne ne le permet pas actuellement. Un essai de directive en la matière fut rejeté en 2005 par le parlement européen. Les opposants à ce projet soulignent qu'un logiciel est composé de milliers d'idées et que le fait de breveter chaque idée indépendamment deviendrait un véritable casse-tête chinois pour les développeurs. Il faudrait se poser la question à chaque ligne de code de l'existence ou non d'un brevet protégeant la fonctionnalité développée. Le rejet du projet de loi en 2005 n'a pas pour autant réglé la question ; certes, la convention de Munich qui déclare le logiciel non brevetable reste applicable, mais les actions de lobbying ont repris des deux côtés et se poursuivent actuellement.

L'office européen des brevets, en contradiction avec la convention de Munich, aurait déjà accordé 40 000 brevets sur le logiciel.

La notion de licence « libre »

Protéger des logiciels par une licence d'utilisation n'est pas une idée nouvelle. L'idée originale fut de créer des modèles de licence permettant de protéger les libertés ci-dessus. La licence publique générale GNU (*GNU GPL* pour *GNU General Public License*), définie par Richard Stallman il y a plus de vingt ans, n'en est qu'un exemple parmi d'autres.

La notion de copyleft¹ « est une façon de rendre un programme ou toute autre œuvre libre, et qui requière que toutes les versions modifiées et étendues du programme soient libres également ».

Les licences de logiciel libre sont reconnues sur le plan juridique. Elles permettent que la plupart du temps les conflits soient réglés à l'amiable, le droit d'auteur classique s'appliquant dans le cas où la licence serait rejetée par une des parties. On ne compte pas plus de 40 conflits juridiques par an sur la licence GPL qui est la plus utilisée. Beaucoup de conflits sont réglés à l'amiable et très peu de cas portés devant la justice.

Il y a en fait moins de licences dans le monde libre que dans le monde propriétaire, où l'on peut rencontrer plusieurs versions différentes de licence pour un même logiciel. 95% des logiciels libres sont couverts par 3 licences : GPL, BSD et LGPL. L'assemblage juridique des licences est ainsi plus facile.

Les avantages du « libre »

L'indépendance

Liberté et indépendance font bon ménage. Le libre permet de ne pas être lié à un fournisseur particulier qui serait trop lié à un état, une religion, un parti politique. Cette indépendance est un critère de sécurité important. Benoît nous cita le cas d'un logiciel antivirus qui aurait été développé par l'église de scientologie et qui pouvait susciter la méfiance de ses utilisateurs : le logiciel ne collectait-il pas des données sur leur ordinateur, à des fins inavouées ?

Dans une période de guerre économique permanente, l'accès au code source offre une transparence rassurante. Le développement d'un logiciel par des acteurs multiples dont les intérêts peuvent être contradictoires permet une multiplicité de points de vue et apporte finalement une garantie de confiance.

Les entreprises travaillant dans le secteur de la sécurité nationale ou celui des armées, les télécoms, la bourse apprécient particulièrement le logiciel libre pour cet aspect sécuritaire.

Le logiciel libre permet d'échapper au contrôle du marché par un acteur particulier. C'est aussi une garantie de pérennité vis-à-vis des entreprises : on se souvient des problèmes posés à certaines entreprises par l'arrêt d'OS2 par IBM. Avoir plusieurs auteurs, donc plusieurs détenteurs des droits, permet aussi la limitation des risques.

La sécurité

Des comparaisons entre le nombre de failles de sécurité des logiciels propriétaires et des logiciels libres sont souvent communiquées par les éditeurs. Ce critère est jugé non pertinent par Benoît : faut-il prendre en compte le nombre ou la gravité des failles ? Comparer les failles locales ou les failles

distantes ? Ces batailles de chiffres n'ont pas grand sens et il est clair qu'elles ne peuvent être menées que par un observatoire scientifique indépendant.

Sur le plan de la sécurité, il est plus intéressant de regarder le processus de correction des erreurs et d'amélioration des logiciels. Qui peut corriger un logiciel propriétaire ? Seulement l'éditeur, et, ajoute Benoît, il faut qu'il en ait envie, que cela représente un intérêt économique suffisant pour lui. En attendant que la correction soit réalisée, un système peut rester vulnérable assez longtemps.

Dans le cas du logiciel libre, le processus est plus souple : les bugs triviaux peuvent être corrigés directement en interne, ce qui permet par exemple d'éviter de couper un réseau en attendant le correctif. Les intervenants potentiels sont plus nombreux, les délais sont raccourcis.

Benoît nous cita le cas de la base de données inter-base, précédemment propriétaire puis devenue libre. Ce passage au libre permit de détecter un trou de sécurité (des mots de passe en dur dans la base) qui n'avait jamais été vu jusqu'alors. Les logiciels libres ont certes des bugs et des problèmes de sécurité comme les autres logiciels ; la différence c'est qu'on peut les corriger plus facilement...

Autre exemple de problème de sécurité induit par des pratiques propriétaires : l'installation par Sony-BMG d'un logiciel de type « rootkit » sur le PC d'un utilisateur afin d'empêcher la copie de CD. Il s'agit d'une intrusion caractérisée sur le système de l'utilisateur. Le logiciel en question introduisait une faille de sécurité sur la machine de l'utilisateur. Un correctif fourni pour le désinstaller introduisait un autre trou de sécurité... Sony-BMG s'en est sorti avec une image un peu écornée...

Menaces juridiques

La gestion des droits numériques ou DRM est l'autre grande menace vis-à-vis du libre. Pour résumer ce point, le terme est devenu synonyme de mesures techniques de protection, ou plutôt de contrôle, informatique mises en œuvre par les éditeurs de logiciels, de musique, et plus généralement de tout support numérique. Les partisans du « libre » rejoignent les défenseurs de la protection de la vie privée et de la confidentialité des données personnelles, en opposant ces pratiques de défiance au monde de confiance qu'ils défendent. En synthèse : boîte noire contre protocoles ouverts. Nous vous conseillons de consulter les nombreux articles et blogs sur les projets de loi et directives autour du droit d'auteur aux USA et de la DADVSI en France.

Les questions qui se posent

Le libre est-il professionnel ?

Ne serait-ce pas du bricolage d'amateur ? Utiliser des logiciels libres est-il vraiment professionnel ?

¹ Le terme copyleft est un jeu de mots faisant référence au copyright traditionnel (opposition right (droit) et left (gauche)) et à l'expression copy left (copie laissée ou copie autorisée). Inventé par Don Hopkins, il a été repris par Richard Stallman comme principe de base de la licence GNU.

À cette question un peu provocatrice, la réponse fut sans appel : « pas d'internet sans logiciel libre »; $\frac{2}{3}$ des serveurs web utilisent des logiciels libres, $\frac{3}{4}$ des serveurs de noms de domaine. Les serveurs de courriel et serveurs web sont majoritairement des logiciels libres. De même, les postes clients utilisent de plus en plus des logiciels libres comme mozilla, firefox, thunderbird ou openoffice. Le déploiement actuel des blogs et des wikis se fait également sur la base d'outils libres.

Les États et administrations des grandes villes migrent vers des solutions libres. On peut citer plusieurs ministères français, la gestion des impôts, ou encore la gendarmerie nationale qui migre vers open office.

Qui sont les Développeurs ?

Deuxième question : qui peut bien travailler pour la communauté de façon désintéressée ?

Ici la réponse fut plus complexe et plus nuancée. Professionnels et amateurs participeraient chacun pour une part. Contrairement à des idées reçues, les contributions viendraient majoritairement du monde professionnel. IBM aurait injecté un milliard de dollars, en particulier dans les développements Apache, et Sun de son côté dans openoffice.

Les entreprises s'intéressent aux logiciels libres pour leurs qualités techniques en matière de robustesse, de sécurité et pour leur pérennité. L'indépendance vis-à-vis des fournisseurs est aussi un critère important. Plusieurs entreprises peuvent associer leurs moyens pour faire des développements qu'elles ne pourraient envisager chacune de façon indépendante.

La part des universitaires dans le développement du libre est jugée réduite par notre orateur. Au début de l'histoire du libre la part des bénévoles et des passionnés était plus importante, aujourd'hui la part des entreprises est croissante.

Un autre phénomène favorisant la montée du libre est une particularité du fonctionnement des administrations aux USA : tous les logiciels développés dans le domaine public, de même que toutes les données, appartiennent aux citoyens.

En France, on observe la mutualisation inter-régions de logiciels de marché publics ou encore le regroupement d'administrations pour des développements d'intérêt commun.

Comment gèrent-ils la configuration ?

Le logiciel libre comme le propriétaire fait l'objet d'une gestion des versions. Chaque branche est gérée par une équipe. Des aiguillages (« fork ») peuvent se produire : à un moment une nouvelle version apparaît en parallèle pour viser un autre objectif. C'est par exemple le cas lorsqu'il faut choisir entre sécurité et performance. Des voies parallèles apparaissent sur certains systèmes d'exploitation suivant que l'objectif est plutôt la portabilité ou la sécurité. Il peut également y avoir des éclatements

liés à des conflits de personne ou à des intérêts nationaux divergents.

Le fait que plusieurs projets soient développés en parallèle sur un même sujet est perçu comme un avantage : ceci permet des échanges entre les projets parallèles.

Les limites du logiciel libre ?

La question des limites du « libre » fut posée. Pouvaient-on aller jusqu'à développer des PGI en libre ?

Benoît Sibaud nous assura qu'il n'y avait pas de limite au développement du logiciel libre.

Les participants soulignèrent toutefois quelques problèmes pratiques rencontrés sur le terrain. Le cas d'un outil de CMS¹, utilisé pour le développement d'un site Internet, fut cité comme exemple de projet abandonné et logiciel non maintenu, obligeant ses utilisateurs, incapables d'en assurer eux-mêmes la maintenance, à changer de solution.

Le logiciel libre s'est d'abord développé au début des années 80 dans les couches basses des systèmes d'exploitation, puis dans les interfaces graphiques, les navigateurs internet, les logiciels de courrier, pour atteindre des fonctions de plus en plus compliquées au fur et à mesure que l'on remonte dans les couches.

Les couches les plus hautes, celles des PGI, commencent aujourd'hui à être attaquées.

La question de maîtrise des délais fut également posée : si vous avez besoin d'un module ou d'une évolution pour une date donnée, il ne faut pas s'en remettre au bon vouloir de la communauté mais probablement s'intégrer dans la communauté et participer aux développements, voire les financer.

Quel modèle économique ?

Le logiciel libre peut être diffusé sur internet mais peut aussi être vendu à des clients. De nombreuses SSII vendent du logiciel libre. Ce logiciel, une fois acheté, peut être redistribué gratuitement par l'acheteur.

L'avantage pour l'acheteur est de pouvoir disposer d'une personnalisation. Il est toujours possible de payer quelqu'un pour réaliser les modifications de code dont on a besoin.

Face au logiciel libre, les sociétés de services ont modifié leurs modèles économiques. Elles ont souvent développé des branches ou pôles logiciel libre. On constate toutefois que les SSII ne sont pas de gros contributeurs en matière de libre. Elles font toutefois travailler des indépendants en sous-traitance pour répondre aux besoins de leurs clients.

Les commerciaux de sociétés de services travaillant au forfait sont confrontés désormais à des demandes de baisse de prix de la part de leurs clients, lorsque des logiciels libres sont utilisés dans les développements. Cela semble justifié. Le modèle écono-

¹ Content Management System

mique du libre repose sur un principe simple : il faut bien rémunérer le développement, mais ne le faire payer qu'une fois. L'alternative utopique pour assurer un développement gratuit serait de ne compter que sur les développeurs bénévoles ; mais dans ce cas on ne pourrait exiger aucun délai d'aucun développeur. Un bénévole ne développera que les fonctionnalités sexy qui l'intéressent.. On touche ici à la limite du bénévolat : le respect des délais se paye !

Processus de développement et qualité logiciel ?

Quel processus de développement est utilisé pour le logiciel libre ? Les méthodes agiles, comme par exemple Extreme programming sont-elles adaptées ? La réponse fut un peu évasive... Il n'y a pas de règle ; chacun fait comme il l'entend, suivant ses habitudes, son propre modèle. Chaque équipe choisit sa méthode de développement ; il n'y a pas forcément de méthode unanimement reconnue comme la meilleure. Suivant le cas, les développements sont faits en « modèle fermé », les versions n'étant révélées qu'une fois prêtes, ou en « modèle ouvert », les développements en cours étant alors annoncés de façon transparente.

Cette réponse ne nous renseigne pas trop sur les pratiques des développeurs du libre. Qui fait quoi, entre les développeurs et les utilisateurs ? Le rôle des utilisateurs ne doit pas être sous-estimé. Les plus actifs peuvent proposer des idées. Leurs suggestions sont précieuses. Ils « remontent » les bugs et peuvent demander des fonctionnalités. Il fut également question de la contribution des traducteurs : les traductions répondent aux besoins et envies des utilisateurs ; par exemple mozilla firefox fut traduit en breton avant de l'être en français. Plusieurs logiciels ont été traduits en Espéranto. Certains États lancent des traductions pour des raisons politiques évidentes : ainsi l'Afrique du sud a tenu à traduire certains logiciels utilisés par ses administrations dans les 11 langues du pays. Les gros projets sont généralement traduits dans une cinquantaine de langues.

Le processus n'est donc pas défini de façon particulière sur un projet libre. On note toutefois l'importance du travail coopératif via internet. Sans réseau, pas de logiciel libre !

Comment ça marche ?

La question n'est pas simple : comment des gens qui ne se connaissent pas peuvent-ils travailler ensemble ? Déjà, quand on se connaît, c'est compliqué !

Quel est le mode d'organisation ? Qui décide ? Benoît nous expliqua que, s'il n'y avait pas de chef, il y avait quand même quelquefois un « dictateur bienveillant », sorte de chef de projet respecté par les autres. Par exemple, dans le cas du noyau de Linux les décisions de Linus Torvalds sont respectées par les autres contributeurs.

Il est essentiel qu'il n'y ait pas de coupure entre les décideurs et les techniciens compétents : on notera que dans le monde propriétaire cette coupure est souvent opérée par la fonction marketing.

Tout cela n'est-il pas un peu fragile ?

Ces communautés de développeurs ne sont-elles pas fragiles ? Lorsque le fondateur passe à autre chose, pour des raisons personnelles ou professionnelles, qu'est-ce qui se passe ?

N'y a-t-il pas des personnes indispensables ? Benoît tenta de nous rassurer : lorsqu'un projet grossit, il y a en principe plusieurs personnes compétentes. C'est d'ailleurs pareil dans le monde propriétaire, voire pire, puisque les équipes sont plus petites.

Et qu'en est-il de la question de l'engagement ? Qui s'engage à quoi dans le domaine du volontariat ? Là, une réponse spontanée vint de la salle : le monde propriétaire n'est pas plus respectueux de ses engagements ; chez les éditeurs d'AGL il y a eu aussi des projets arrêtés et des clients laissés sur le carreau ! Certaines entreprises rachètent leur concurrent pour le laisser mourir. Au moins dans le monde du libre un projet peut être repris s'il intéresse quelqu'un.

Et les tests ?

La question du processus de test fut également soulevée.

Le libre rassemble beaucoup de testeurs. Il offre un ban d'essai incomparable. Mais il ne s'agit pas de « vrais » tests : les utilisateurs utilisent ; ils ne testent pas. Tout le monde peut tester les mêmes fonctionnalités. Par exemple, les failles de sécurité ne sont pas visibles de l'utilisateur ordinaire, seules les fonctionnalités sont testées.

Les exigences de qualité font l'objet de tests plus complets sur les gros projets financés.

L'assurance qualité n'est donc pas garantie dans tous les cas. Chacun doit se porter responsable du niveau de qualité qu'il souhaite obtenir.

La plupart du temps il existe des systèmes de suivi de bugs publics permettant à chacun de se faire une idée des points faibles d'un logiciel. Des logiciels de recherche de bugs existent également dans le monde du libre.

Il faut bien être conscient que le besoin de plus de qualité se paye : les utilisateurs de l'armée font réaliser des audits de sécurité qui profitent ainsi à toute la communauté des utilisateurs.

Les tests ont ainsi pris de l'importance depuis que les logiciels libres sont utilisés pour des besoins de plus en plus stratégiques. Des projets européens sont en cours sur le thème de la qualité du logiciel libre.

Les méthodes de test ne sont pas forcément les méthodes classiques : les tests de non-régression sont difficiles à exécuter de façon systématique lorsque les versions se succèdent à un rythme rapide. Seuls de gros projets comme le développement du compilateur GCC (compilateur le plus utilisé de la planète) font l'objet de tests de régression

systematiques. Chacun sait que les développeurs n'aiment pas tester, que ce soit dans le monde du libre ou dans le monde propriétaire. Il est important d'automatiser les tests.

De fait, les couches basses des systèmes d'exploitation sont les mieux testées et les Unix libres sont aujourd'hui plus fiables que les Unix propriétaires. Le nombre d'éditeurs de systèmes d'exploitation diminue ; les Unix propriétaires sont en voie de disparition, remplacés progressivement par les Unix libres.

Et le respect des normes ?

Liberté est-il synonyme d'anarchie ? Les logiciels libres respectent-ils les normes et standards ? La question était facile et la réponse ne se fit pas attendre : oui, les logiciels libres respectent les normes et standards en usage, qu'elles proviennent de l'ISO, de l'IETF ou du W3C. Et ils respectent encore plus les protocoles du Web que ne le font les logiciels propriétaires. Internet explorer par exemple ne se conforme pas à l'ensemble de ces standards, alors que les navigateurs libres s'efforcent de le faire.

Le respect des standards et l'interopérabilité ne sont pas de l'intérêt du propriétaire. Il est clair en effet

qu'un comportement protectionniste crée de la captivité. Les entreprises en position dominante ne sont pas motivées à respecter les standards.

Conclusion

Quel est donc l'avenir du logiciel libre ?

- d'un côté une croissance continue et très rapide qui ne devrait que s'accélérer ;
- d'un autre côté des menaces plus juridiques que techniques en provenance d'éditeurs et lobbies qui font pression pour durcir la législation et introduire la brevetabilité des logiciels.

Laquelle de ces deux tendances l'emportera ? Difficile à dire aujourd'hui, tant les intérêts sont opposés entre l'« informatique de confiance » et l'« informatique de défiance ».

Souhaitons que ce soit l'intérêt des usagers qui l'emporte. ▲

martine.otter@adeli.org