

# Introduction au logiciel libre

Jérôme Gleizes

DANS **MULTITUDES 2000/1 n° 1**, PAGES 161 À 165

ÉDITIONS **ASSOCIATION MULTITUDES**

ISSN 0292-0107

DOI 10.3917/mult.001.0161

Date de mise en ligne : 01/05/2008

Article disponible en ligne à l'adresse

<https://shs.cairn.info/revue-multitudes-2000-1-page-161?lang=fr>



Découvrir le sommaire de ce numéro, suivre la revue par email, s'abonner...  
Scannez ce QR Code pour accéder à la page de ce numéro sur Cairn.info.



# Introduction au logiciel libre

PAR JÉRÔME GLEIZES

Un logiciel est un ensemble d'instructions donné à un ordinateur, écrit dans un langage de programmation. Deux niveaux se confondent : c'est à la fois le programme, l'ensemble du code source et la forme exécutable, dite compilée dudit programme. Pour l'utilisateur, seul le second niveau importe. Un logiciel libre (*free software*) est un logiciel fourni avec son code source (i.e. le programme du logiciel), donnant le droit à toute personne de l'utiliser, le copier, le modifier librement et le distribuer (y compris dans ses versions modifiées). Il peut être commercialisé dans une version exécutable mais il est toujours possible d'obtenir le code source. Souvent, il est disponible par téléchargement depuis un site Internet ou par copie d'un CD-ROM.

Certains développeurs de logiciels libres tel Eric S. Raymond à travers l'*Open Source Initiative* (OSI) préfèrent l'expression « logiciel source ouverte » (*open source software*) au lieu de celle de « logiciel libre » (*free software*). En voulant supprimer une ambiguïté sémantique — « free » signifie aussi bien « gratuit » que « libre » -, ils négligent ce qui fait la spécificité du logiciel libre, la liberté de modifier et de redistribuer sans contrainte un programme libre en se focalisant sur la seule accessibilité aux codes sources. Mais, ils laissent planer une nouvelle ambiguïté avec l'expression « logiciel ouvert » (*open software*). Cette expression a été créée dans les

années quatre-vingt par Sun, HP, DEC, IBM, et d'autres sociétés pour souligner la compatibilité technique entre différents logiciels et non la disponibilité du code. Cet appauvrissement du concept de logiciel libre amène certains et notamment les journalistes à classer inopportunément comme libres des logiciels qui ne le sont pas : les logiciels semi-libres, c'est-à-dire des logiciels mélangeant des modules libres et propriétaires (non-disponibilité du code) tel que Netscape Communicator, et même des logiciels non-libres, c'est-à-dire des logiciels dont l'usage est limité tels les logiciels sous licence SCSL (*Sun Community Source License*).

Le concept de logiciel libre a été créé par Richard Stallman dans les années quatre-vingt pour des raisons éthiques. En 1971, quand Richard Stallman démarre sa carrière dans un laboratoire d'intelligence artificielle du MIT (*Massachusetts Institute of Technology*), les informaticiens hackers et les chercheurs des plus grandes universités américaines utilisent essentiellement des logiciels libres. Les sociétés informatiques distribuent des logiciels propriétaires et parfois des logiciels libres. Mais progressivement, elles imposent des logiciels propriétaires, allant même jusqu'à privatiser du code libre. La création de la société Sun dans les années quatre-vingt, profitant des failles juridiques du droit du logiciel, symbolise ce mouvement, en privatisant les logiciels du monde Unix.

Pendant, il est toujours possible de trouver une application libre. Ainsi un jour, Richard Stallman est confronté à un problème avec une imprimante de chez Xerox. Il lui est impossible d'ajouter une fonction supplémentaire au programme car il ne possède pas le code source. Un collègue chercheur possède bien celui-ci, mais il ne peut le transmettre car il a signé un contrat de non-diffusion avec la société Xerox. Richard Stallman pense alors que cette personne n'a pas respecté l'éthique de la communauté des informaticiens « en faisant la promesse immorale de refuser aux autres ce qu'il voulait pour lui-même. » Au lieu de continuer sa carrière d'informaticien dans l'université et signer des contrats de non-diffusion, il décide de démissionner pour se consacrer à l'écriture d'un système d'exploitation libre et éviter ainsi que l'université ne dépose des brevets sur ses logiciels au détriment des utilisateurs. À cette époque, la communauté des hackers est décimée, incapable de résister aux propositions financières des sociétés produisant uniquement des logiciels propriétaires.

Au début des années quatre-vingt, presque tous les logiciels libres sont devenus des logiciels propriétaires. Ceux qui en détiennent les droits interdisent et empêchent ainsi toute coopération entre utilisateurs. En réponse à cela, en 1983, Richard Stallman lance l'idée d'un projet intitulé *GNU*, susceptible de rétablir l'esprit coopératif qui prévalait dans la communauté des hackers et des chercheurs. Ce terme *GNU* est en fait un acronyme récursif comme cela se fait communément dans le milieu hacker : *GNU's Not Unix*. En janvier 1984, l'idée se concrétise et en octobre 1985, il crée la *Free Software Foundation* (FSF) afin de mieux structurer le projet, de développer une communauté d'utilisateurs actifs et de s'autofinancer en vendant des CD-ROM de logiciels libres ou en recevant des dons.

Le projet *GNU* a permis de développer un système complet de logiciels libres. Trois libertés spécifiques y sont défendues. « Premièrement, la liberté de copier le programme et de le donner. Deuxièmement, la liberté de modifier le programme sans aucune contrainte, grâce à un accès complet au code source. Troisièmement, la liberté de distribuer une version améliorée et ainsi aider la communauté. »

La première étape du projet *GNU* a été de construire un système d'exploitation du même nom. Ce logiciel est l'élément central d'un ordinateur, lui seul permet l'utilisation d'un ordinateur, indépendamment des programmes d'application, et la gestion des périphériques (imprimantes, lecteur de disque, etc.). Un système d'exploitation se compose d'un noyau mais inclut également des compilateurs, des éditeurs, des formateurs de texte, des logiciels de courrier, etc. L'écriture d'un système d'exploitation complet et cohérent est un travail essentiel pour garder sa liberté et son autonomie face aux logiciels propriétaires. Cela a pris de nombreuses années. Les initiateurs du projet, essentiellement Richard Stallman, ont décidé de rendre le système d'exploitation compatible avec le système d'exploitation Unix parce que celui-ci avait déjà fait ses preuves et que cette compatibilité rendait plus facile le passage d'Unix à GNU.

Au début des années quatre-vingt-dix, tous les composants majeurs sont écrits, excepté le noyau. Un noyau libre, Linux, est alors développé par un finlandais, Linus Torvalds. Cet événement affirme l'importance d'Internet dans la diffusion et le développement du logiciel libre. Les listes de diffusion et les forums électroniques permettent alors de décou-

pler et d'internationaliser la coopération entre informaticiens mais aussi entre utilisateurs passifs et actifs, les premiers testant les logiciels écrits par les seconds. La combinaison du noyau Linux avec les logiciels GNU crée un système d'exploitation complet : un système GNU basé sur Linux (GNU/Linux). Richard Stallman estime à 20 millions, le nombre d'utilisateurs de systèmes GNU/Linux, incluant les distributions Debian, Red Hat, Mandrake, SuSe, etc.

Cependant, le projet GNU ne se limite pas aux systèmes d'exploitation. Il concerne aussi les logiciels applicatifs (tableur, traitement de texte, etc.) Il essaie également de fournir des logiciels pour les utilisateurs qui ne sont pas des experts en informatique grâce à des interfaces graphiques ergonomiques mais aussi des jeux. Il concerne aussi la documentation et les manuels d'utilisation des logiciels libres. Pour Richard Stallman, c'est un autre élément essentiel du développement du logiciel libre. Un logiciel libre ne peut exister sans un manuel d'utilisation libre. Il doit être libre, c'est-à-dire appropriable par tous afin de le compléter, de l'améliorer.

Mais jusqu'où le logiciel libre peut-il aller ? Il n'y a pas de limites, excepté les lois régissant le commerce et la propriété intellectuelle (droits d'auteurs et brevets). L'objectif ultime est de fournir des logiciels libres pour effectuer toutes les tâches informatiques – et de cette manière rendre obsolète le logiciel propriétaire. Mais aujourd'hui, la possibilité de breveter des logiciels est le principal danger pour les logiciels libres car même si le code reste libre d'accès, sa modification et sa diffusion pourraient être impossibles sans l'accord du propriétaire.

Au logiciel libre du projet GNU sont associées des licences spécifiques pour faire face à un détournement de la philosophie du logiciel libre : La GPL (*General Public License*), licence qui spécifie les conditions de distribution des logiciels et la LGPL (*Library General Public License*) pour les bibliothèques de sous-programmes. Ces licences sont basées sur le principe du droit d'auteur (*copyright*) mais donnent ensuite l'autorisation légale de dupliquer, distribuer et/ou modifier le logiciel (notion de *copyleft*, ou « gauche d'auteur » selon la traduction de Richard Stallman). Il existe cependant des licences de logiciel libre qui n'interdisent pas la privatisation et la fermeture des codes des versions modifiées (licence *BSD*). Dans ce cas, le logiciel se rapproche de la catégorie des logiciels du domaine

public. À la différence d'un programme du domaine public, un programme libre peut appartenir à ses auteurs (*copyright*) mais en aucun cas ses auteurs et ceux qui vont l'acquérir ne peuvent refuser la diffusion des codes sources initiaux et ceux des versions successives (*copyleft*).

Mais actuellement, nous assistons à un détournement de plus en plus important de la philosophie du logiciel libre par l'introduction de nouvelles licences (*Netscape Public License (NPL)*, *Mozilla Public License*, etc.). Ainsi, la distinction sémantique entre « free software » et « open source software » est aussi idéologique. Sur le site personnel de Eric S. Raymond, défenseur de l'*open source software* et aussi auteur de la NPL, nous pouvons trouver un FAQ (*Frequently Asked Questions*) sur le libérarisme. Pour ce dernier et pour l'OSI, la liberté est la valeur fondamentale alors que pour Richard Stallman et la FSF, l'égalité et la fraternité importent aussi, ce qui finalement est une opposition classique entre une pensée libérale et une pensée de gauche. ■