

# Introduction au logiciel libre

## Introduction

On entend souvent parler de logiciels libres ou open source mis en opposition à ceux propriétaires et privés. Mais qu'est-ce que c'est exactement, et pourquoi s'en soucier ?

## Un peu d'histoire

L'informatique était essentiellement accessible aux chercheurs au début des années 1970. Ils partageaient couramment entre eux les **codes sources** des programmes qu'ils développaient. C'est, par analogie, une sorte de « recette », qui une fois « cuisinée » nous donnera notre « gâteau » (Figure 1). Remplacer ici la « cuisine » par un processus nommé compilation, qui transforme la « recette » en **binaire** – suite de zéros et d'uns –, exécutable par un ordinateur, mais illisible pour un programmeur – un « cuisinier » pour continuer l'analogie. La possession du code source est donc impérative pour adapter, améliorer ou réparer un logiciel.

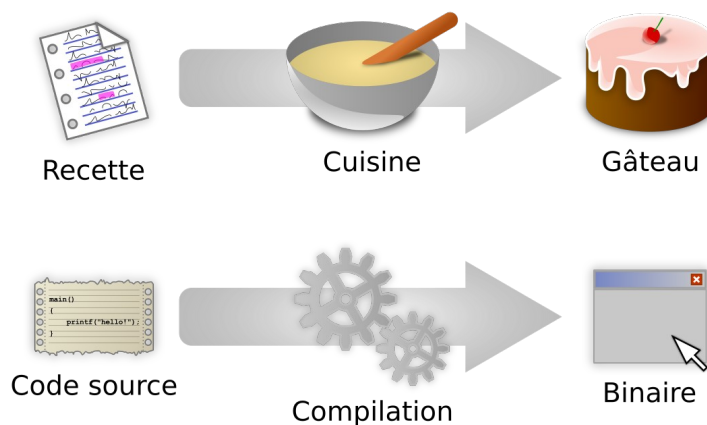


Figure 1: Illustration de l'analogie

L'échange des sources entre programmeurs bénéficiait à tous ; chacun pouvait adapter les logiciels à ses besoins, tout en participant à leur amélioration, en partageant les modifications apportées.

À savoir qu'à l'époque, les entreprises du secteur informatique basaient leurs revenus sur la fourniture de matériel. Mais c'est à cette période qu'elles ont commencé à s'orienter vers la vente de logiciels. Cette commercialisation a entraîné une forte protection des codes sources sous couvert du secret d'entreprise.

C'est en 1979 que Richard Matthew Stallman, étudiant au MIT, un prestigieux institut de recherche américain, se heurte à ce nouveau problème. Ayant participé à cette culture de partage, il est choqué qu'un collègue refuse de lui fournir le code source d'un logiciel de pilotage d'une imprimante, qu'il voulait corriger. Mais celui-ci étant développé par Xerox, son détenteur a signé une clause de confidentialité avec l'entreprise.

Cette anecdote témoigne de la disparition de la culture de Stallman. Pour la préserver, il fonde en 1985 avec des collègues la **Free Software Foundation** (FSF, Fondation pour le Logiciel Libre), dont la notion de logiciel libre est issue. Il publie ensuite son logiciel Emacs sous la nouvelle licence de l'organisation, la **GNU General Public License**, assurant la disponibilité du code source.



Figure 2: Le GNU (prononcé "gnou" en anglais), mascotte de la FSF et de leurs licences

[Tête de GNU contrasté](#) par Aurélio A. Heckert sous licence [CC BY-SA 2.0](#) © 2003, Free Software Foundation, Inc.

## Libertés sous licence

Pour qu'un logiciel soit considéré comme libre, selon la définition de la Free Software Foundation, il doit offrir les droits de :

- L'utiliser comme on le souhaite, sans conditions ;
- L'étudier, le modifier et l'adapter à ses besoins, ceci **requiert la diffusion du code source** ;
- Redistribuer des copies ;
- Partager ses modifications.

Ces libertés sont garanties par une **licence**, document administratif joint au programme par son développeur. Les droits sont protégés par un copyright, détenu par l'auteur. Il faut donc comprendre qu'un copyright n'interdit pas systématiquement l'utilisation libre d'une œuvre ; il assure que ces conditions soient celles définies par l'auteur.

Les codes sources dans le domaine public sont aussi considérés comme libres. La différence avec ceux sous licence est que l'auteur n'a plus de droits sur l'œuvre, il ne peut donc s'opposer à une quelconque utilisation de celle-ci. A savoir qu'une œuvre entre dans le domaine public seulement si son auteur cède ses droits, ou 70 ans après son décès.

Les logiciels libres sont mis en opposition à ceux dits **privateurs** ou **propriétaires** – par confusion, comme vu, une œuvre libre peut rester protégée. Ceux-ci n'offrent pas lesdites libertés, et les codes sources sont rarement fournis. On peut citer par exemple Microsoft Windows, Adobe Photoshop et Google Chrome.

Certaines licences libres sont dites **copyleft** (littéralement « gauche d'auteur », jeu de mot sur copyright), elles obligent toute version dérivée d'un programme à offrir les mêmes libertés que celui-ci. C'est le cas des licences GNU General Public License (GNU GPL) et Creative Commons Attribution – Partage dans les Mêmes

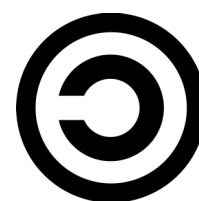


Figure 3: Symbole du copyleft

Conditions (CC BY-SA). Les licences **permissives** dont la MIT et Apache ne sont quant à elles pas copyleft ; les versions altérées peuvent être diffusées sous licences privatives.

Il faut aussi comprendre qu'il est autorisé de vendre un logiciel libre. Mais la fourniture obligatoire du code source, et le droit de quiconque de redistribuer, à son tour, librement – donc gratuitement –, font qu'une rentabilisation se fait rarement par ce biais.

D'autres médias peuvent être sous licence libre. Par exemple la Licence Art Libre (LAL ou FAL, Free Art License) est copyleft et adaptée aux œuvres d'art, tandis que la GNU Free Documentation License (GNU FDL, aussi copyleft) est adaptée aux documents texte – physiques ou non.

## Enjeux

### Transparence

Nombres de tâches et de décisions sont aujourd'hui déléguées à des logiciels – souvent dits algorithmes –, donnant un pouvoir important à leurs détenteurs. Les missions des programmes sont extrêmement variées, allant de la gestion de courriers électroniques, à l'orientation en étude supérieure (Parcoursup), en passant par l'envoi et l'arbitrage de démarches administratives (création d'entreprise, demande d'allocations, déclaration d'impôts...).

Tout aspect de la vie étant, à différentes échelles, informatisé, il devient impératif de contrôler ces programmes. Les logiciels privés n'offrant pas leurs codes sources, protégés comme secrets industriels – ou d'Etat –, rendent impossible cet encadrement. L'étude d'un programme est extrêmement complexe sans son code, la compréhension de ses rouages reste alors l'affaire de ses seuls développeurs. Le citoyen est donc privé du droit de regard – qui lui est pourtant légitime – sur l'un des plus importants pivots de décision politique, économique et social.

Les monopoles d'entreprises comme les GAFAM (acronyme désignant Google, Amazon, Facebook, Apple et Microsoft) donne d'autant plus de gravité à ce manque de transparence ; les données des utilisateurs sont traitées sans qu'ils en sachent le fonctionnement ou les finalités, les entreprises se déchargeant sous couvert de conditions d'utilisations larges. Bien que l'opacité des programmes ne soit pas le seul problème dans ce cas, il reste l'un des plus importants.

### Savoirs communs

De nombreuses connaissances et technologies – au-delà de l'informatique – sont aujourd'hui bloquées par divers dispositifs légaux tels les brevets ou la propriété intellectuelle. Ces verrous offrent à leurs détenteurs le monopole dans leurs secteurs. Bien que nous abordions principalement l'informatique, on voit que la problématique s'applique à tous champs techniques et scientifiques.

Les utilisateurs sont alors piégés dans l'écosystème des entreprises. Ces dernières peuvent abuser de leur position ; documentation non fournie créant une dépendance au service, dépréciation d'un produit forçant le passage à une nouvelle génération (obsolescence programmée), brevetage d'une technique concurrente bloquant son usage...

La notion de libre propose une alternative en créant des **communs**. En donnant les droits décrits dans le chapitre Libertés sous licence sur les savoirs et techniques, on garantit à tous un accès égal à

ceux-ci. En découle la création de communauté autour de projets, ou chaque membre participe à son développement.

Certaines entreprises du secteur informatique choisissent aussi des stratégies plus respectueuses de leurs clients. Distribuant leurs logiciels sous licences libres, elles vendent alors leurs expertise et services. C'est le cas de l'entreprise Red Hat et de son système d'exploitation Red Hat Entreprise Linux (RHEL), profitant et participant à la version communautaire Fedora, et laissant l'alternative CentOS utiliser ses sources.

## Sécurité

Les logiciels privateurs ne fournissant pas leurs codes sources, il est presque impossible de connaître leurs fonctionnements internes. Par conséquent, on ne peut savoir si ceux-ci sont exempts de **failles de sécurité**, exploitable par des **malwares** – appelés virus à tort, qui n'en est qu'un type –, ou de fonctions malveillantes. Par exemple en 2017, le rançongiciel WannaCry s'est propagé mondialement sur les machines sous Windows XP de Microsoft, à cause d'une technique développée par la NSA (National Security Agency, organisme gouvernemental américain de renseignement) ayant fuitée.

Au contraire, le code source public des logiciels libres permet l'audit indépendant de ceux-ci, sans l'aval des développeurs. Aussi, leur développement souvent communautaire, rend difficile l'insertion furtive d'une fonction malveillante. Et grâce à cette organisation, les logiciels libres sont souvent vite corrigés lorsqu'une faille est découverte.

De plus, lors de l'arrêt du développement d'un logiciel privateur, rien ne garantit qu'il sera mis à jour, et donc que les potentielles failles soient corrigées. Tandis que si un logiciel libre est abandonné par son équipe, libre à une autre de reprendre le projet. Le caractère libre d'un programme lui donne donc une bien meilleure pérennité qu'un privateur.

## Et l'open source ?

Le terme open source émerge en 1998, pour rendre attractif le logiciel libre auprès de certains utilisateurs – en particulier des entreprises – en se déchargeant des valeurs politiques et philosophiques. L'Open Source Initiative – créée la même année que le terme –, fait figure d'autorité dans le domaine de l'open source à l'instar de la FSF pour le logiciel libre.

La notion d'open source met la priorité sur la viabilité économique et les qualités techniques ; développement communautaire créant des programmes sûrs et performants, de forts arguments commerciaux et un modèle parfaitement rentable. La FSF promulguant le logiciel libre comme libérateur avant tout, elle rejette le terme.

Bien que la majorité des logiciels libres rentre dans la définition de l'open source selon l'OSI – et vice-versa –, les différentes approches peuvent avoir des répercussions conséquentes. L'open source ne se met pas nécessairement en opposition aux programmes privateurs, la stratégie est juste considérée différente.

Ces divergences se notent, entre autres, par le fait que les licences des projets open sources ne sont souvent pas copyleft. Pour exemple Android Open Source Project (AOSP) de Google, qui constitue

la base du système d'exploitation Android, publié sous la licence non copyleft Apache. Cette stratégie assure un quasi-monopole à l'entreprise ; prenant à sa charge la majorité des coûts de développement, les constructeurs d'équipements comme Wiko, Samsung et LG dérivent d'AOSP leurs propres versions privatives, la licence étant permissive. De plus, Google laisse privatur nombres de composants du système.

Tout de même, nombre de projets utilisent les termes libre et open source de manière interchangeable. Les acronymes FOSS (Free and Open Source Software) et FLOSS (Free/Libre Open Source Software) sont parfois utilisés pour regrouper les deux notions.

## Alternatives libres

Ci-dessous une liste non exhaustive de logiciels libres, avec leurs équivalents privés s'il y a.

Fonction	Logiciels privés	Alternatives libres	Description
Suite bureautique	Microsoft Office	<a href="#">LibreOffice</a>	Suite bureautique complète avec éditeur de texte, tableur, mise en page, présentation et base de données. Ce document a été rédigé avec LibreOffice Writer.
Lecteur PDF	Adobe Acrobat	<a href="#">Sumatra PDF</a>	Lecteur de fichier PDF simple et léger.
Navigateur Web	Google Chrome Microsoft Edge Apple Safari	<a href="#">Mozilla Firefox</a>	Convivial et accès sur la protection de la vie privée, personnalisable avec des modules, comme des bloqueurs publicitaires.
Client mail	Microsoft Outlook	<a href="#">Mozilla Thunderbird</a>	Regroupe toutes vos boîtes mails dans une seule interface dédiée, sans passer par votre navigateur Web. Propose de nombreuses fonctions comme un carnet de contact et le filtrage des spams.
Correcteur grammatical	Synapse Cordial	<a href="#">Grammalecte</a>	Là où de nombreux logiciels offrent un correcteur d'orthographe, Grammalecte vient fait de même avec la grammaire. Compatible avec Firefox, Thunderbird et LibreOffice.
Cartographie mentale	OpenGenius Ayoa	<a href="#">Freeplane</a>	Créer des cartes mentales mises en forme exportables en PDF pour les diffuser.
Prise de notes	Evernote	<a href="#">CherryTree</a>	Hiérarchiser vos notes mises en forme dans un fichier qui peut être protégé par mot de passe.
Lecteur multimédia	Microsoft Windows Media Player	<a href="#">VLC</a>	Lit la majorité des formats de vidéos et musicaux, à partir de fichiers ou de disques.
Compression	WinRAR	<a href="#">7-Zip</a>	Ouvre et compresse les archives de multiples formats, dont ZIP, RAR et TAR.
Maintenance système	Piriform CCleaner	<a href="#">BleachBit</a>	Maintiens et nettoie le système d'exploitation et de nombreux logiciels.
Gestionnaire de mots de passe	1Password Dashlane	<a href="#">KeePassXC</a>	Centralise vos identifiants et mots de passe dans une base de données stockée hors ligne.
Personnalisation	Stardock Start10	<a href="#">Open Shell</a>	Retrouvez le menu Démarrer classique sous Microsoft Windows 8 et 10.
Traitement d'images	PhotoFiltre	<a href="#">Photoflare</a>	Modifiez vos photos et créez des graphismes avec ce logiciel conçu spécialement pour les éditions simples et les néophytes.
	Adobe Photoshop	<a href="#">GIMP</a>	Complet et puissant, mais pas nécessairement intuitif, ces capacités vont de la retouche photo au dessin avec tablette graphique.
Dessin vectoriel	Adobe Illustrator	<a href="#">Inkscape</a>	Créer des illustrations, graphiques, diagrammes et mises en page en partant d'opérations

			sur des formes géométriques simples.
Edition audio	Apple GarageBand	<a href="#">Audacity</a>	Mixez, corrigez et éditez facilement vos pistes audios avec ce puissant outil. Il permet aussi l'enregistrement via un microphone.
Musique assistée par ordinateur	Image-Line FL Studio	<a href="#">LMMS</a>	Ce logiciel très complet donne à tous l'accès à la musique assistée par ordinateur professionnelle.
	Steinberg Cubase		
Edition vidéo	Microsoft Windows Movie Maker	<a href="#">OpenShot</a>	Spécialement conçu pour une prise en main rapide, il reste tout de même capable de réaliser des montages complexes.
	Sonic Foundry Vegas Pro	<a href="#">Kdenlive</a>	Complet, il se rapproche des logiciels privés professionnels.
Architecture	Micro Applications 3D Architecte	<a href="#">Sweet Home 3D</a>	Réalisez des plans architecturaux et simulez leurs aménagements.
Modélisation 3D	BrickLink Studio	<a href="#">LeoCAD</a>	Inventez vos propres modèles à partir de briques LEGO.
	Autodesk 3ds Max	<a href="#">Blender</a>	Performant mais complexe, il est particulièrement utilisé pour le jeu vidéo et l'animation 3D. Il permet même l'édition vidéo et inclus un moteur de jeu complet.
Photogrammétrie	Agisoft Metashape	<a href="#">Regard3D</a>	Créer des modèles 3D à partir de séries de photos, sans le besoin d'un scanner dédié.
Moteur de jeu	Unity Yoyo Games Game Maker	<a href="#">GDevelop</a>	Créez vos propres jeux vidéos 2D avec cet outil conçu pour être utilisé facilement, même sans expérience préalable.
Conception assistée par ordinateur	Autodesk AutoCAD	<a href="#">LibreCAD</a>	Dessinez des plans et autres schémas techniques en 2D.
		<a href="#">OpenSCAD</a>	Créez des modèles 3D grâce à son puissant langage de programmation.
		<a href="#">FreeCAD</a>	Conceptualisez plans et pièces techniques pour une réalisation matérielle.