

Cours Logiciels Libres

Introduction, modalités pratiques et histoire du logiciel libre

Théo Zimmermann

Université de Paris & Inria

Vendredi 21 janvier 2022

Qui suis-je ? Quelle est mon expertise ?

- Membre de l'équipe de développement de Coq, un “assistant de preuve” (voir le cours Preuves Assistées par Ordinateur), distribué sous **licence libre** et développé de manière **ouverte**.
 - Coq compte + de 200 contributeurs et + de 30 mainteneurs.
- Fondateur de l'organisation communautaire Coq-community.
 - Coq-community maintient une soixantaine de projets (+ de 30 mainteneurs).
- Ma recherche a pour but de comprendre et d'améliorer la manière dont les développeurs **collaborent** pour maintenir du logiciel libre.
 - Ma thèse portait sur ce sujet dans le cadre de Coq.

C'est quoi un logiciel libre ?

C'est quoi un logiciel libre ?

Un *logiciel libre*, c'est un logiciel distribué sous une *licence* qui autorise ceux qui le reçoivent à l'**utiliser**, en **étudier le code source**, le **modifier** et le **redistribuer** (inchangé ou modifié).

Le contraire d'un logiciel libre est un *logiciel propriétaire*.

C'est quoi un logiciel libre ?

Un *logiciel libre*, c'est un logiciel distribué sous une *licence* qui autorise ceux qui le reçoivent à l'**utiliser**, en **étudier le code source**, le **modifier** et le **redistribuer** (inchangé ou modifié).

Le contraire d'un logiciel libre est un *logiciel propriétaire*.

Il s'agit donc de garantir des *libertés* à l'utilisateur (là où les licences de logiciels propriétaires restreignent l'utilisation, l'accès au code source et la redistribution).

Mais au-delà de la question des libertés, cela permet aussi un autre modèle de développement, collaboratif et ouvert, qu'on désigne plus souvent sous le terme d'*open source*.

Vous utilisez déjà des logiciels libres

Vous utilisez déjà des logiciels libres

Directement :

- Votre système d'exploitation si vous êtes sous GNU/Linux,
- Votre navigateur web si c'est Firefox ou Chrome,
- Une partie de votre téléphone si vous êtes sous Android,
- Votre éditeur de texte si c'est Emacs, Vim, VsCode ou Atom,
- Le compilateur de votre langage préféré,
- Le gestionnaire de version git,
- etc.

Indirectement (dépendances à des bibliothèques libres) :

- La plupart des sites web sur lesquels vous naviguez,
- La plupart des applications mobiles que vous utilisez.

Pourquoi ce cours est utile pour vous

- En tant que futur·es ingénieur·es.
- En tant que futur·es chercheur·ses.
- En tant que citoyen·nes.
- À titre personnel.

En tant que futur·es ingénieur·es

En tant que futur·es ingénieur·es

- Pour vous faire recruter : votre profil GitHub est votre deuxième CV.
- L'**utilisation de logiciel libre** est essentielle dans presque toutes les entreprises (dans et hors secteur informatique) :
 - Réduction des coûts,
 - Adaptabilité,
 - Durabilité.
- Le logiciel libre peut aussi être moteur de **collaborations** au-delà des frontières de l'entreprise.

En tant que futur·es chercheur·ses

En tant que futur·es chercheur·ses

- Le partage de code source sous licence libre permet de :
 - reproduire les résultats de la recherche,
 - construire à partir des résultats existants,
 - comparer différentes techniques.
- Le logiciel libre est la manière dont les communautés de chercheur·ses collaborent entre équipes et créent des projets majeurs.

En tant que citoyen·nes

En tant que citoyen·nes

- Loi pour une République numérique (2016) : les administrations doivent publier leurs codes et leurs données.
 - <https://code.gouv.fr/> (2021)
- On peut inspecter le code source du calcul des impôts, de TousAntiCovid, etc. (mais ces logiciels ne sont pas développés de manière ouverte).
- Des associations et des entreprises développent du logiciel libre au service de la démocratie participative : voir Code for France, Decidim, etc.

À titre personnel

À titre personnel

Lorsque vous **utilisez** des logiciels libres, vous pouvez :

- en rapporter des bugs,
- regarder et modifier le code source pour corriger des bugs, l'adapter à vos besoins,
- faire partie d'une communauté.

Si vous **développez** un logiciel ou une bibliothèque et que vous souhaitez le partager et recevoir des contributions.

Ce que nous allons y voir

- Histoire (début, grands événements, évolution récente).
- Philosophies et idéologies.
- Aspect juridique (droit d'auteur, copyright et licences libres).
- Comment :
 - Utiliser du logiciel libre.
 - Contribuer à du logiciel libre (→ projet).
 - Maintenir du logiciel libre.
- Gouvernance, gestion de communautés.
- Modèles économiques.

Modalités pratiques

- Cours et TD / TP le vendredi de 16h15 à 18h15 en salle 1008.
- Évaluation : moitié projet / moitié examen.
- Projet : contribuer à un logiciel libre de votre choix, typiquement pour y corriger un bug, et écrire un rapport sur votre expérience (détails à suivre).
- Site du cours : theo.im/libre

Free software, logiciel libre et open source

Ces trois termes veulent fondamentalement dire la même chose, mais en mettant l'accent sur différents aspects.

Free software, logiciel libre et open source

- Le terme “free software” a été popularisé par Richard Stallman et la Free Software Foundation (1985).
- Il met l’accent sur les **libertés** (“Free Software Definition”) :
 - ① Liberté d’utiliser le programme, pour n’importe quel usage.
 - ① Liberté d’étudier comment le programme fonctionne et de le modifier (requiert l’accès aux sources).
 - ② Liberté de redistribuer des copies du programme.
 - ③ Liberté de redistribuer des versions modifiées du programme.
- Son problème est que “free” peut vouloir dire “libre” ou “gratuit”. Les défenseurs du “free software” doivent souvent répéter “Free as in free speech, not as in free beer” pour rappeler que ce n’est pas une question de prix.

Free software, **logiciel libre** et open source

- Le terme “logiciel libre” (“software libre” en espagnol) est la traduction de “free software”.
- Il n’a pas ce problème de double sens.
- Certaines personnes choisissent de l’utiliser en anglais pour éviter la confusion (→ LibreOffice).

Free software, logiciel libre et **open source**

- Le terme “open source” a été popularisé par Eric Raymond et l’Open Source Initiative (1998).
- Il met l’accent sur le **partage** du code permettant la **collaboration ouverte** (“Open Source Definition”).
- Son problème est que beaucoup de gens confondent “publication du code source” et “open source”, alors que le code source peut être publié sous une licence restrictive

Free software, logiciel libre et open source

Pour éviter toute confusion et éviter de prendre parti dans un débat philosophique, certains parlent de FOSS ou FLOSS :
pour **F**ree, (**L**ibre,) and **O**pen **S**ource **S**oftware.

En pratique, la quasi-totalité des logiciels libres sont open source et inversement.

Avant le logiciel libre

- Dans les débuts de l'informatique, **les ordinateurs coûtent très cher** et sont distribués avec des logiciels spécialement conçus pour eux (dont la valeur est secondaire).
- Des programmes sont modifiés, conçus et partagés par leurs utilisateurs, sans règles particulières.
- On finit par considérer que le logiciel a une valeur en soit, peut être vendu et est protégé par le **copyright** (droit d'auteur).
- **Microsoft prend son envol** à travers un accord avec IBM : IBM vend des PC (personal computers) avec le système d'exploitation de Microsoft (MS-DOS), sur lequel ce dernier conserve les droits. L'industrie du logiciel est née.

Walter Isaacson

L'auteur de la biographie de référence de Steve Jobs

Les Innovateurs



Comment un groupe de génies,
hackers et geeks a fait la révolution numérique

9.

Le logiciel

Lorsque Paul Allen s'approcha du kiosque à journaux encombré au milieu de Harvard Square et aperçut l'Altair en couverture du *Popular Electronics* daté de janvier 1975, il fut à la fois enthousiasmé et consterné. Bien qu'il soit emballé par la révélation que l'ère de l'ordinateur individuel avait commencé, il craignait d'arriver trop tard pour participer à la fête. Il posa rageusement ses soixante-quinze cents sur le comptoir, s'empara du magazine et trotta dans la neige fondue jusqu'à la chambre en cité universitaire de Bill Gates, son pote de lycée de Seattle, mordu d'informatique lui aussi, qui l'avait convaincu de décrocher de la fac et de s'installer à Cambridge. « Hé ! ce truc est en train de nous passer sous le nez ! », déclara Allen. Gates se mit à osciller d'avant en arrière, comme cela lui arrivait souvent dans des moments particulièrement intenses. Quand il eut fini de lire l'article, il se rendit compte qu'Allen avait raison. Pendant les huit semaines suivantes, ils se lancèrent dans une frénésie programmatrice qui allait changer la nature commerciale de l'informatique¹.

Figure 2: Les Innovateurs, Chapitre 9

Unix

- Lancé à partir de 1969 chez AT&T (Bell Labs).
- Distribué à de nombreux industriels et universités qui en créent des **variants**.
- Aujourd'hui les descendants d'Unix sont :
 - La famille BSD (Berkeley Software Distribution) :
 - FreeBSD, OpenBSD, NetBSD,
 - Darwin, le noyau (libre) de macOS.
 - À l'origine de la famille de **licences BSD**.
 - GNU / Linux (GNU = GNU's Not Unix)

Le projet GNU

- Lancé à partir de 1983 par Richard Stallman.
- Pour créer un Unix complètement libre.
- Une **collection de projets** divers :
 - GNU Emacs,
 - GCC, glibc et GDB,
 - GNU Bash,
 - GNU Core Utilities (cat, ls, rm, etc.),
 - GRUB,
 - GParted,
 - GIMP,
 - GPG,
 - etc.
- Et la famille de **licences GNU** (GPL, LGPL, AGPL).
- Création de la Free Software Foundation en 1985.

X Window System (X11)

- Lancé à partir de 1984 au MIT.
- Système de fenêtrage très utilisé sous Unix.
- Distribué sous licence X11, à l'origine de la **licence MIT**.
- Géré depuis 2004 par la X.Org Foundation.

Le noyau Linux

- Lancé à partir de 1991 par Linus Torvalds.
- Premier logiciel libre **développé de manière collaborative** sur internet (envoi de patches par newsgroups, puis par e-mail).
- Sous **licence GNU GPL 2.0** depuis 1992.
- A inspiré l'essai "The Cathedral and the Bazaar" à Eric Raymond (à l'origine du mouvement "open source").
- Création de la **Linux Foundation** en 2000, qui collecte aujourd'hui plus de 6 millions de dollars annuels de la part de ses plus de 1000 entreprises membres.

Les premières distributions GNU/Linux

- Debian :
 - Lancée à partir de 1993 par Ian Murdock.
 - L'une des toutes premières distributions Linux
 - Entièrement **communautaire** et presque entièrement libre.
- Red Hat (distribution et entreprise) :
 - Distribution lancée à partir de 1994 par Marc Ewing.
 - L'une des toutes premières entreprises à produire une distribution Linux et en **vendre la maintenance**.
 - Version communautaire : Fedora.
 - L'entreprise Red Hat est **rachetée par IBM** en 2019 pour 34 milliards de dollars.

Apache : un fork communautaire

- NCSA HTTPd était l'un des tout premier serveur web et le plus utilisé dans le monde de 1993 à 1995. (Développé à l'Université d'Urbana-Champaign par Robert McCool).
- Le projet ayant cessé d'être maintenu, ces utilisateurs décident d'en produire un **fork** en agrégeant des patchs qui circulaient.
- Le nom est un jeu de mot accidentel avec "a patchy server".
- **Serveur web le plus utilisé** dans le monde de 1995 à 2016 (conurrencé depuis par Nginx).
- Donne lieu à la création de l'**Apache Foundation** et de la **licence Apache 2.0**.

Mozilla

- Netscape était le navigateur dominant à partir de 1995, jusqu'à l'arrivée d'Internet Explorer en 1997, qui écrase le marché en étant **distribué avec Windows**.
- En 1999, l'entreprise **Netscape libère le code de son navigateur** qui devient Mozilla (puis Mozilla Firefox), influencé par l'essai "The Cathedral and the Bazaar" d'Eric Raymond.
- Firefox parvient à être un **concurrent sérieux d'Internet Explorer** jusqu'à ce que Google Chrome arrive et écrase tout à son tour.
- Création de la **Mozilla Foundation** en 2003. Aujourd'hui également connue pour d'autres projets, notamment le langage Rust.
- À l'origine de la **licence MPL 2.0**.

StarOffice, OpenOffice, LibreOffice

- Star Division, une entreprise allemande publiant la suite StarOffice, est rachetée en 1999 par Sun Microsystems.
- En 2000, **Sun Microsystems libère le code** de la suite qui devient OpenOffice.
- Création en 2005 du format standard OpenDocument.
- **Oracle rachète Sun Microsystems** en 2009 et se désengage d'OpenOffice.
- Un **fork communautaire** est créé : LibreOffice.
- OpenOffice est cédé par Oracle à la Fondation Apache.

Android

- Système d'exploitation open source **basé sur Linux**, développé par Google et publié à partir de 2007.
- Permet de **concurrencer Apple** et l'iPhone (plus de 70% du marché).
- N'est pas développé de manière collaborative ou ouverte, mais la licence **autorise les fabricants** de smartphones à en produire leur **version dérivée**.
- En pratique, la plupart des smartphones sous Android sont **loin d'être libres** car ils incluent aussi les Google Mobile Services (Google Play) qui sont propriétaires.

SourceForge

- Lancé en 1999, c'est l'une des premières **forge** de logiciels libres.
- Permet aux projets d'avoir un dépôt de code, un bug tracker, un wiki, une mailing list ou un forum, une page de téléchargement, etc.
- Simplifie le partage de **petits projets open source**, les petits forks de projets non maintenus, etc.
- Plus de 500 000 projets y ont été hébergés.
- Mais en 2013, le site a perdu en crédibilité à cause de certaines pratiques douteuses, et beaucoup de projets ont alors migré vers GitHub.

git et GitHub

- Les gestionnaires de versions traditionnels (CVS, SVN) étaient **centralisés** (nécessitent un dépôt central, un accès à ce dépôt, voir l'utilisation de verrous) : peu adaptés au modèle de **collaboration ouverte** de l'open source.
- Le projet Linux bascule sur un gestionnaire **décentralisé mais propriétaire** : BitKeeper. La licence permet de l'utiliser pour développer des logiciels libres, mais son utilisation provoque une controverse et la licence finit par être révoquée.
- Linus Torvalds crée **git** à partir de 2005 comme alternative libre à BitKeeper.
- **GitHub est lancé en 2008** par quatre développeurs en Californie, popularise le concept de **pull request** et l'utilisation de **forks de développement**.

Les gestionnaires de paquets

- Les débuts (avant les forges) : des archives pour **partager des extensions** (Emacs Lisp Archives, CTAN, CPAN, CRAN, etc.)
- Les **distributions Linux** se construisent autour de leur gestionnaire de paquet (dpkg en 1994 pour Debian, qui sert toujours de brique de base pour APT).
- Apparitions des **écosystèmes de paquets** modernes construits autour de gestionnaires de paquets spécifiques dans les années 2000 (RubyGems, Cabal) et 2010 (npm, pip, Composer, opam, Cargo, etc.)
- Explosion du nombre de **dépendances** des projets logiciels. Conséquences médiatiques quand des tout petits paquets font disparaissent (LeftPad).

Microsoft et l'open source

- Open Letter to Hobbyists (1976) : Bill Gates se plaint du “**vol**” de ses logiciels.
- Steve Balmer compare Linux à “un **cancer** qui s’attache à tout ce qu’il touche” (2001).
- Microsoft commence à contribuer à Linux en 2009, publie des langages de programmation open source (F# en 2005, **TypeScript** en 2012).
- Satya Nadella écrit “Microsoft <3 Linux” en 2014 (Microsoft fait tourner Linux dans Azure depuis 2012), Microsoft devient l’un des plus gros **sponsor de la Linux Foundation** en 2018.
- **VsCode** est publié en open source en 2015.
- Le **Windows Subsystem for Linux** est introduit en 2016.
- Microsoft rachète **GitHub** en 2018.

Le logiciel libre a-t-il gagné ?

- On pourrait croire que oui si on regarde les **outils utilisés par les développeurs** sur leurs ordinateurs (éditeurs, gestionnaires de version, compilateurs, bibliothèques).
- Mais si on regarde en termes de **libertés des utilisateurs finaux**, on en est loin !
- La plupart des applications web et mobiles **dépendent de logiciels libres** mais ne donnent pas de libertés à l'utilisateur.
- GitHub n'est pas libre ! (Même si des alternatives libres, comme GitLab existent.)
- Et Microsoft Word continue aussi à dominer le marché face à LibreOffice.