Introduction aux Logiciels Libres et Open Source

Amine Brikci-Nigassa

amine.brikcinigassa@univ-tlemcen.dz

Département d'Informatique - UABTlemcen

Licence en informatique 1^{re} année - 2025-2026

Présentation de la matière

- Unité d'enseignement UEM 111
- Volume horaire semestriel (14 semaines) :
 21h
- Volume horaire hebdomadaire :
 - Cours : 1h30
 - Autres : 2h (travail personnel)
- Coefficient: 1
- Crédits : 2
- Mode d'évaluation : Examen final (100%)
- Connaissances préalables recommandées : Connaissances générales en informatique.

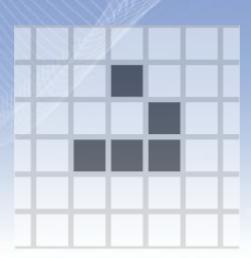
Contenu de la matière

- Chapitre 1 : Technologies de l'information
 - Définitions
 - Outils
 - Matériels (Hardware)
 - Ordinateurs
 - Réseaux de communication
 - Périphériques
 - Dispositifs mobiles
 - Puces intelligentes
 - Logiciels (Software)
 - Systèmes d'exploitation
 - Applications métiers
 - Logiciels de bureautique
 - Applications

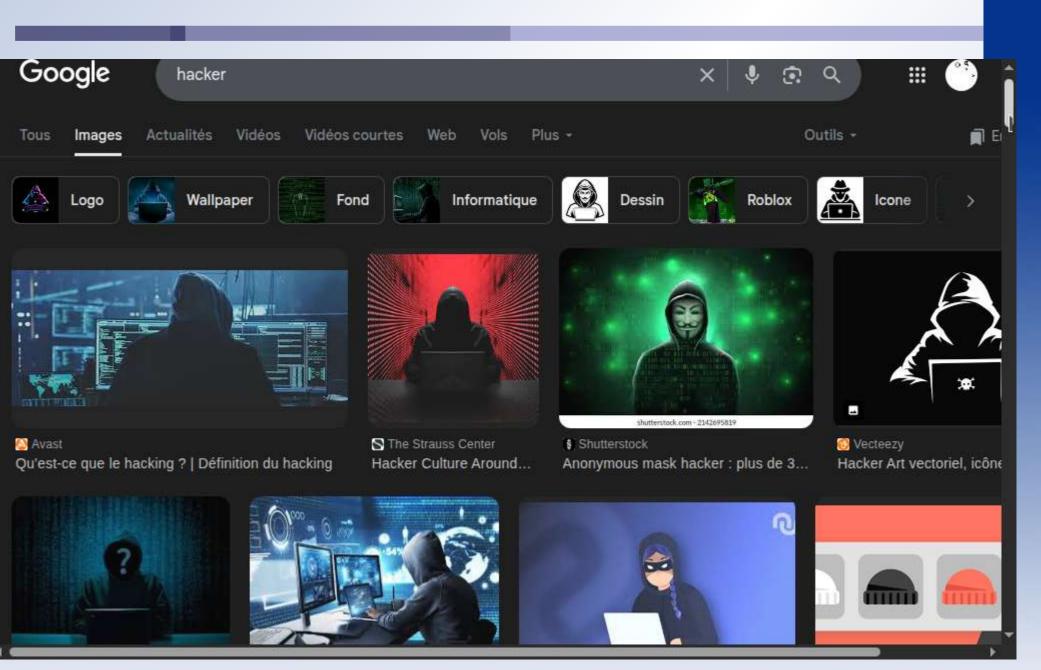
Contenu de la matière

- Chapitre 2 : Outils Open source
 - Introduction
 - Historique
 - Avantages / inconvénients
 - Licences
 - Environnement de développement
 - Introduction à GNU/Linux
 - Introduction aux éditeurs de code
 - Bureautique
 - Suite bureautique LibreOffice
 - Collaboration
 - Stockage et partage
 - Contribuer à un projet open source

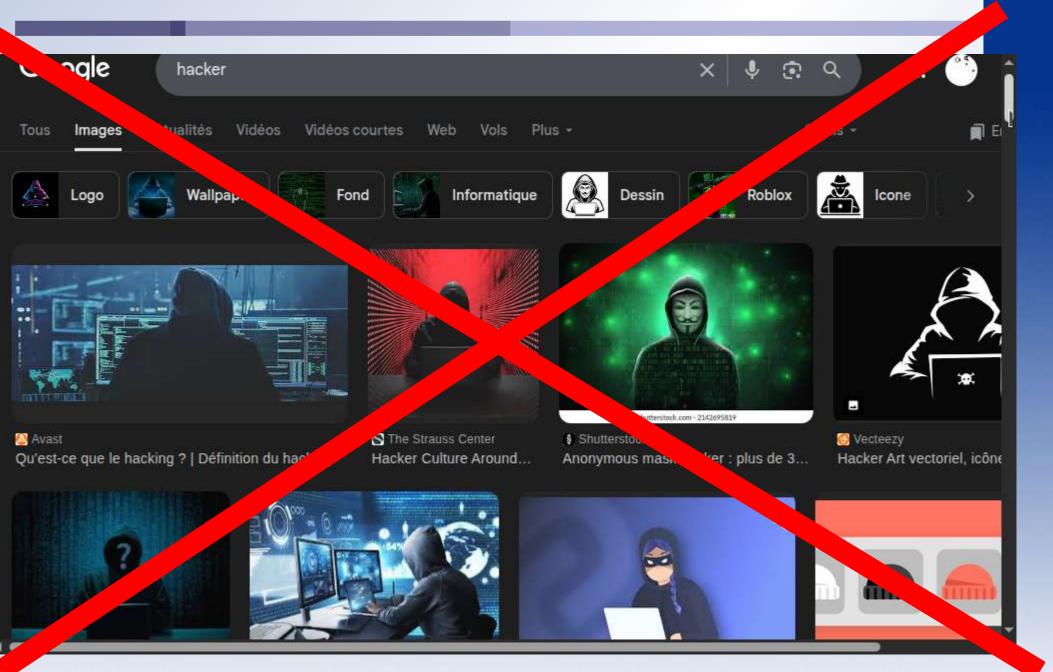




Qu'est-ce qu'un hacker?



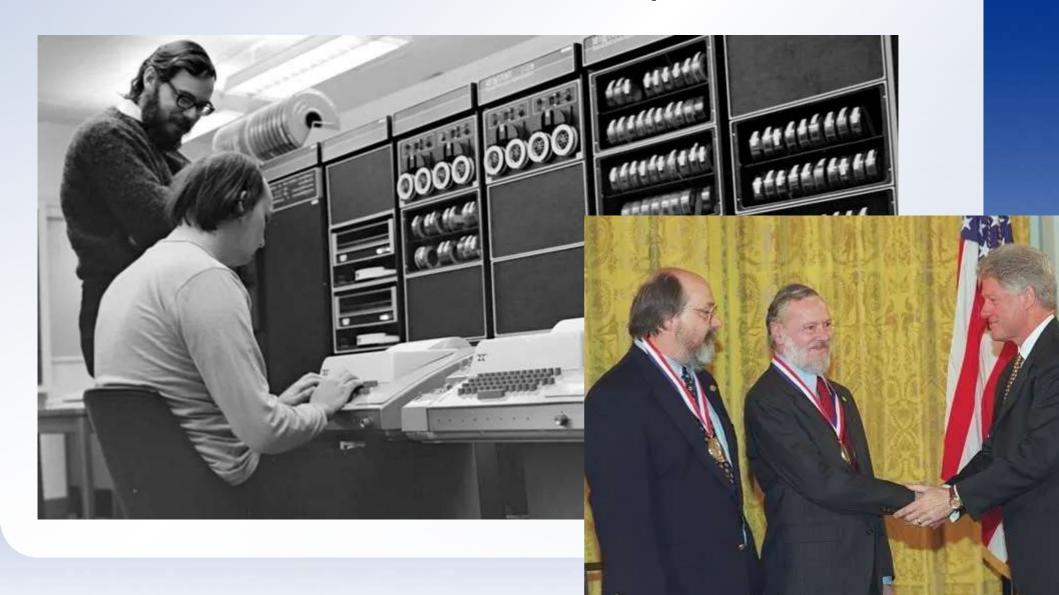
Qu'est-ce qu'un hacker?



John McCarthy



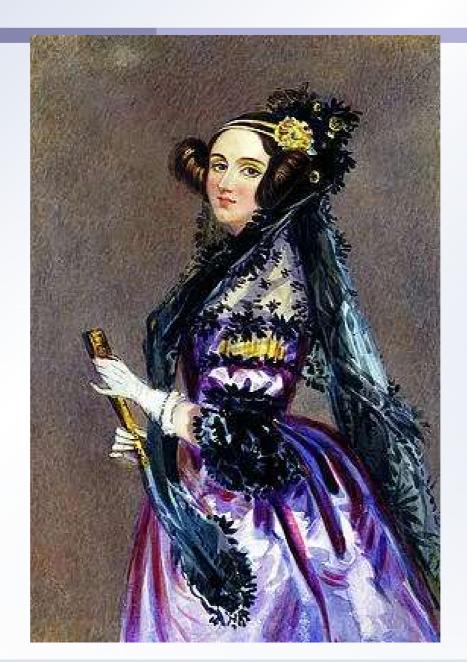
Dennis Ritchie & Ken Thompson



Steve Wozniak



Ada Lovelace



Grace Hopper



Dorothy Vaughan



Hacker: définitions

- Personne qui prend plaisir à comprendre de façon approfondie le fonctionnement interne d'un système, en particulier des ordinateurs et réseaux informatiques. – (définition de la RFC 1983)
- « Il existe une communauté, une culture partagée, de programmeurs expérimentés et de spécialistes des réseaux, dont l'histoire remonte aux premiers mini-ordinateurs multiutilisateurs, il y a quelques dizaines d'années, et aux premières expériences de l'ARPAnet. Les membres de cette culture ont créé le mot « hacker ». Ce sont des hackers qui ont créé l'Internet. Ce sont des hackers qui ont fait du système d'exploitation Unix ce qu'il est de nos jours. Ce sont des hackers qui font tourner les newsgroups Usenet et le World Wide Web. » — Eric Raymond : Comment devenir un hacker

Culture hacker

 Laboratoire d'Intelligence Artificielle du MIT, 1970



Les valeurs de la culture hacker

- Curiosité
- Motivation
- Sens du défi
- Partage
- Sociabilité
- Entraide, solidarité
- Liberté
- Savoir et compétence
- Respect

Compétences du hacker

- Apprendre à programmer :
 - Algorithmique (raisonnement)
 - Langages de programmation :
 - Python
 - C
 - LISP
 - Java, etc.
- Maîtriser les systèmes d'exploitations :
 - Famille des Unix
 - Installer/utiliser une distribution de GNU/Linux (Ubuntu, Kali, ...)
 - Travailler avec la ligne de commande (terminal)
- Réseaux, Internet, Web, HTML...
- Anglais : comprendre, écrire (messages, identificateurs et même commentaires)

Le système UNIX

- Système d'exploitation multitâche multiutilisateur
- Développé au début par Ken Thompson et Dennis Ritchie et l'équipe du laboratoire Bell en 1969 sur mini-ordinateurs
- Modulaire : composé de plusieurs petits programmes (chacun a une fonction simple → Philosophie Unix)
- Devenu portable une fois réécrit en C (1973) → Développement collaboratif : universités, centres de recherche...
- Naissance de plusieurs variantes : famille des Unix (BSD, AIX, SunOS...)

Commerce des logiciels : la menace

- L'esprit de solidarité du hacker attaqué : licences et clauses de confidentialité
- Appât du gain → apparition de restrictions
- Exemple : la « lettre ouverte aux amateurs » de Bill Gates (An Open Letter to Hobbyists) en 1976

February 3, 1976

An Open Letter to Hobbyists

To me, the most critical thing in the hobby market right now is the lack of good software courses, books and software itself. Without good software and an owner who understands programming, a hobby computer is wasted. Will quality software be written for the hobby market?

Almost a year ago, Paul Allen and myself, expecting the hobby market to expand, hired Monte Davidoff and developed Altair BASIC. Though the initial work took only two months, the three of us have spent most of the last year documenting, improving and adding features to BASIC. Now we have 4K, 8K, EXTENDED, ROM and DISK BASIC. The value of the computer time we have used exceeds \$40,000.

The feedback we have gotten from the hundreds of people who say they are using BASIC has all been positive. Two surprising things are apparent, however. 1) Most of these "users" never bought BASIC (less than 10% of all Altair owners have bought BASIC), and 2) The amount of royalties we have received from sales to hobbyists makes the time spent of Altair BASIC worth less than \$2 an hour.

Why is this? As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?

Is this fair? One thing you don't do by stealing software is get back at MITS for some problem you may have had. MITS doesn't make money selling software. The royalty paid to us, the manual, the tape and the overhead make it a break-even operation. One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3-man years into programming, finding all bugs, documenting his pro-

« most of you steal your software »

« the thing you do is theft. »

BASIC (less than 10% of all Altair owners have bought BASIC), and 2) The amount of royalties we have received from sales to hobbyists makes the time spent of Altair BASIC worth less than \$2 an hour.

Why is this? As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?

Is this fair? One thing you don't do by stealing software is get back at MITS for some problem you may have had. MITS doesn't make money selling software. The royalty paid to us, the manual, the tape and the overhead make it a break-even operation. One thing you do do is prevent good software from being written. Who can afford to do professional work for nothing? What hobbyist can put 3-man years into programming, finding all bugs, documenting his product and distribute for free? The fact is, no one besides us has invested a lot of money in hobby software. We have written 6800 BASIC, and are writing 8080 APL and 6800 APL, but there is very little incentive to make this software available to hobbyists. Most directly, the thing you do is theft.

What about the guys who re-sell Altair BASIC, aren't they making money on hobby software? Yes, but those who have been reported to us may lose in the end. They are the ones who give hobbyists a bad name, and should be kicked out of any club meeting they show up at.

I would appreciate letters from any one who wants to pay up, or has a suggestion or comment. Just write me at 1180 Alvarado SE, #114 Albuquerque, New Mexico, 87108. Nothing would please me more than being able to hire ten programmers and deluge the hobby market with good software.

Bill Gates General Partner, Micro-Soft

Richard Stallman





Naissance de la Free Software Foundation

- Richard Matthew Stallman ("RMS"):
 hacker au MIT AI Lab depuis 1971 (18 ans, 1^{re} année universitaire)
- Incident de l'imprimante
- Licences propriétaires : licences « privatrices »
- 1984 RMS démissionne du MIT et crée :
 - Le Projet GNU (GNU's Not UNIX)
 - La F.S.F. (1985)
- Objectifs :
 - Tous les logiciels doivent être libres
 - Respecter la liberté de l'utilisateur
 - Retour de la Culture Hacker



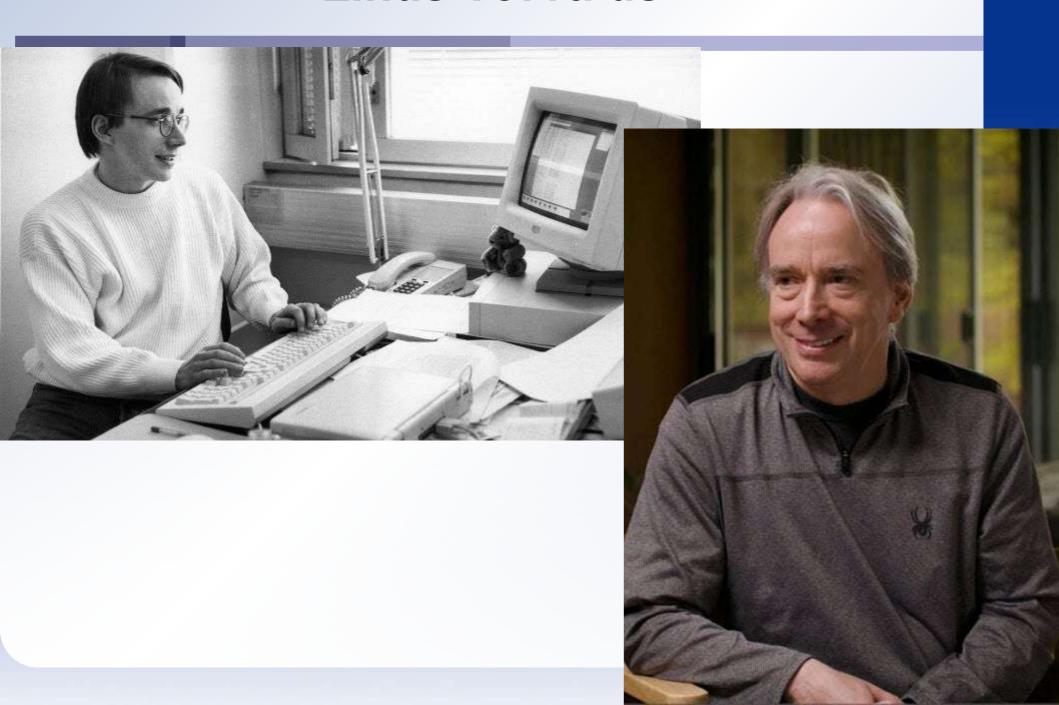
Le système d'exploitation GNU

- Clone de UNIX ⇒ modulaire :
 - 1^{er} programme : GNU Emacs (éditeur de texte)
 - Vendu 150\$: Free = Libre ≠Gratuit
 - GCC (GNU C compiler→GNU compiler collection)
 - glibc (bibliothèque standard C)
 - GDB (GNU debugger)
 - Bash (interpréteur de commande : shell)
 - Intégration de logiciels libres existants :
 - Composition de documents : TeX
 - Fenêtrage graphique : système X-Window
 - ... Gnome (environnement de bureau), etc.
- Licence libre : GNU GPL (General Public License)

Le noyau Linux

- Noyau de GNU ("The Hurd") : en retard (pas encore terminé)
- 1991 Linus Torvalds, étudiant finlandais (21 ans) complète GNU ⇒ Linux
- Linux est seulement un noyau
 - Rôles : gère le matériel (processeur, mémoire, périphériques), les processus (programmes), la communication...
 - Ne fonctionne pas seul
- RMS rappelle : le système d'exploitation, c'est GNU+Linux (ou GNU/Linux), pas Linux !
 - Oublier GNU ⇒ risque d'oublier son objectif (moral, éthique)

Linus Torvalds





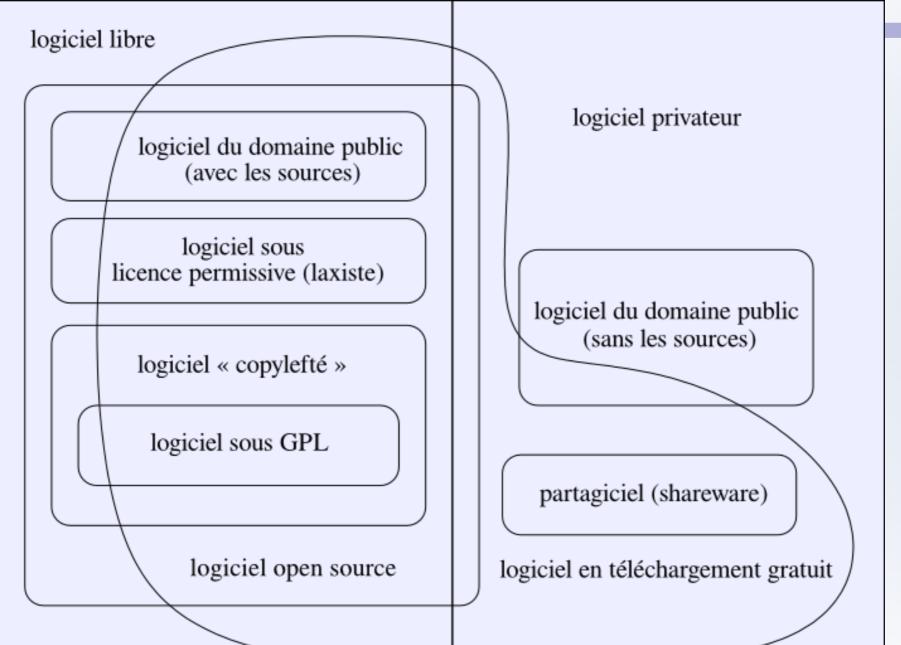
Autres systèmes d'exploitation libres

- De type UNIX :
 - Famille des BSD (Berkeley Software Distribution)
 - FreeBSD
 - OpenBSD
 - NetBSD
 - DragonFly BSD
 - Darwin (base de macOS et iOS d'Apple)
 - illumos
 - Android (noyau = Linux)
 - GNU/Hurd (instable)
 - Redox
- FreeDOS (clone de MSDOS)

•

Open Source Initiative

- 1998 : Open Source Initiative
 - Techniquement : Open Source = Libre
 - Principes Open Source ≠ Principes Libres :
 - Open source : méthodologie de développement (pratique)
 - Logiciel libre : mouvement social (éthique)
 - FSF: Open Source ⇒ Libre (ou Libre/Open Source)



Source: www.gnu.org/philosophy/categories.fr.html

- Logiciels Libres ⇒ 4 libertés fondamentales :
 - 0. Liberté d'utilisation (∀ usage)
 - 1. Liberté de modifier, d'adapter *
 - 2. Liberté de distribuer le logiciel
 - 3. Liberté de distribuer les modifications *

* Libertés 1 et 3 ⇒ ∃ code source Libre ≠ gratuit

- Logiciels du domaine public :
 - pas de ©
 - désistement explicite de l'auteur ou plus de 50 ans après sa mort
- Logiciels privateurs (ou propriétaires): non Libres
- Logiciels copyleftés :
 - Copyleft = gauche d'auteur = ©
 - Libre avec restriction : ne pas changer la licence dans les dérivés (dite « virale » par les opposants)
 - ex. de licences : GPL (@fort), LGPL (@faible)

- Logiciels Libres non copyleftés :
 - licence Libre permissive (pas de restriction)
 - − ≃ domaine public (même problème)
 - ex. de licences : BSD, X11
- Logiciels Open Source :
 - Logiciels Open Source ≃ Logiciels Libres
 - Différences dans les principes
 - Open Source = collaboration, efficacité, performances (aspect *Liberté* négligé)

Logiciels privés :

- appartiennent à la société qui les utilisent
- compatibles avec la philosophie du Libre
- c'est la plus grande catégorie
- Logiciels commerciaux :
 - peuvent être Libres ou non (Libre ≠ Gratuit)
 - l'économie du Libre est plus centrée sur les
 Services (maintenance, support, formation...)

Sharewares et Freewares :

- Non Libres!
- Shareware (Partagiciel) version gratuite d'essai (limite de temps) ex: WinRAR
- Freeware (Gratuiciel) ≠ Free Software (Libre): non modifiable, souvent pas de code source

- GNU General Public License (GPL)
 - avec Copyleft
 - l'une des plus utilisées
 - recommandée par la FSF



- Licences BSD et X11
 - 2 licences permissives, sans copyleft
 - risque de perte de liberté
 - le piège X Window : des constructeurs distribuent des versions adaptées propriétaires
 - déconseillées par la FSF (mais pas par l'Open Source Initiative)

- GNU Lesser General Public License (LGPL)
 - Précédemment appelée Library GPL
 - Créée pour les bibliothèques (glibc, gtk...) mais utilisée aussi pour les applications (ex: KWrite/Kate)
 - Faible copyleft (lesser = amoindrie) : permet
 l'intégration de la bibliothèque dans des logiciels
 propriétaires

- GNU Free Documentation License (GNU FDL)
 - Résout le problème des manuels de LL
 - exemple : Borland Interbase 6 / FireBird
 - Utilisation étendue à d'autres ouvrages (encyclopédies, manuels scolaires, tutoriels...)
 - exemple : Wikipédia, l'encyclopédie libre et gratuite



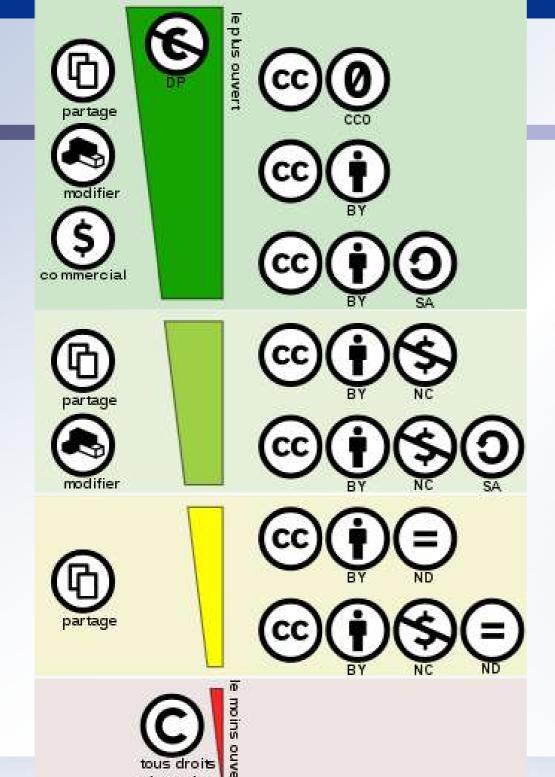
Les licences Creative Commons



- Créées pour les œuvres artistiques (images, vidéo, textes, musiques...) librement redistribuables
- Options combinées ⇒ 6 contrats
- La FSF prévient du danger de confusion : seules 2 des 6 licences sont libres

Les 6 premières licences Creative Commons

Paternité	CC-BY	BY:			libre sans copyleft
Paternité Pas de Modification	CC-BY-ND	BY:		\equiv	non libre
Paternité Pas d'Utilisation Commerciale Pas de Modification	CC-BY-NC-ND	BY:	(=	non libre
Paternité Pas d'Utilisation Commerciale	CC-BY-NC	BY:	(« semi-libre »
Paternité Pas d'Utilisation Commerciale Partage des Conditions Initiale l'Identique	s à CC-BY-NC-SA	BY:	(③	« semi-libre »
Paternité Partage des Conditions Initiale l'Identique	s à CC-BY-SA	BY:		③	libre avec copyleft (Wikipédia)



Les 7 licences Creative Commons

La licence CC-Zero (CC0) est apparue en 2009 pour permettre de renoncer aux droits d'auteur

Source : Wikipédia CC BY-SA 4.0

Cycle de développement des LL/OS

- Style « Bazar » : (Eric S. Raymond La Cathédrale et le Bazar)
 - Cycle itératif
 - « Publiez tôt, publiez souvent. Et soyez à l'écoute de vos clients » (RERO: Release Early, Release Often...)
 - 1^{res} versions simples mais fonctionnelles
 - Nombreuses versions successives => conformité, réajustement aux besoins
 - Exemple typique du modèle Bazar : le noyau Linux
 - "Loi de Linus": « Given enough eyeballs, all bugs are shallow. » Étant donné un ensemble de bêtatesteurs et de co-développeurs suffisamment grand, chaque problème sera rapidement isolé, et sa solution semblera évidente à quelqu'un.