

Les aspects juridiques du logiciel libre

Pierre FICHEUX, CTO Open Wide/OS4I
pierre.ficheux@openwide.fr

Février 2010

Copyright (c) 2010 Pierre Ficheux < pierre.ficheux@gmail.com >

Permission vous est donnée de copier, distribuer et/ou modifier ce document selon les termes de la Licence GNU Free Documentation License, Version 1.1 ou ultérieure publiée par la Free Software Foundation ; sans aucune section inaltérable; sans texte de première page de couverture; sans texte de dernière page de couverture.

Une copie de cette Licence est incluse dans la section appelée GNU Free Documentation License de ce document et peut être consultée à l'adresse www.gnu.org/copyleft/fdl.html.

- La confusion vient de la signification anglaise
 - free = libre
 - free = gratuit
- Différents types de logiciels
 - Le *freeware* ou graticiel: gratuit mais sources non disponibles, pas forcément de licence (abandon de la « paternité » du code)
 - Le *shareware*: sources non disponibles, coût modique, licence souvent propriétaire
 - Le *logiciel libre*: sources DISPONIBLES, licence *open source*, notion de gratuité « décorrelée » (cas de Linux)

- C'est le sujet qui nous intéresse !
- Pour en finir avec les idées reçues :
 - Le logiciel libre n'est pas forcément gratuit
 - La migration vers du logiciel libre n'est JAMAIS gratuite => le « coût du changement »
 - Les licences, même libres doivent être respectées
 - Ces licences ne sont pas incompatibles avec la vente de logiciel (embarqué ou non) ni de matériel
- Par contre il est vrai que les choses ne sont pas toujours claires (le prix de la liberté !)

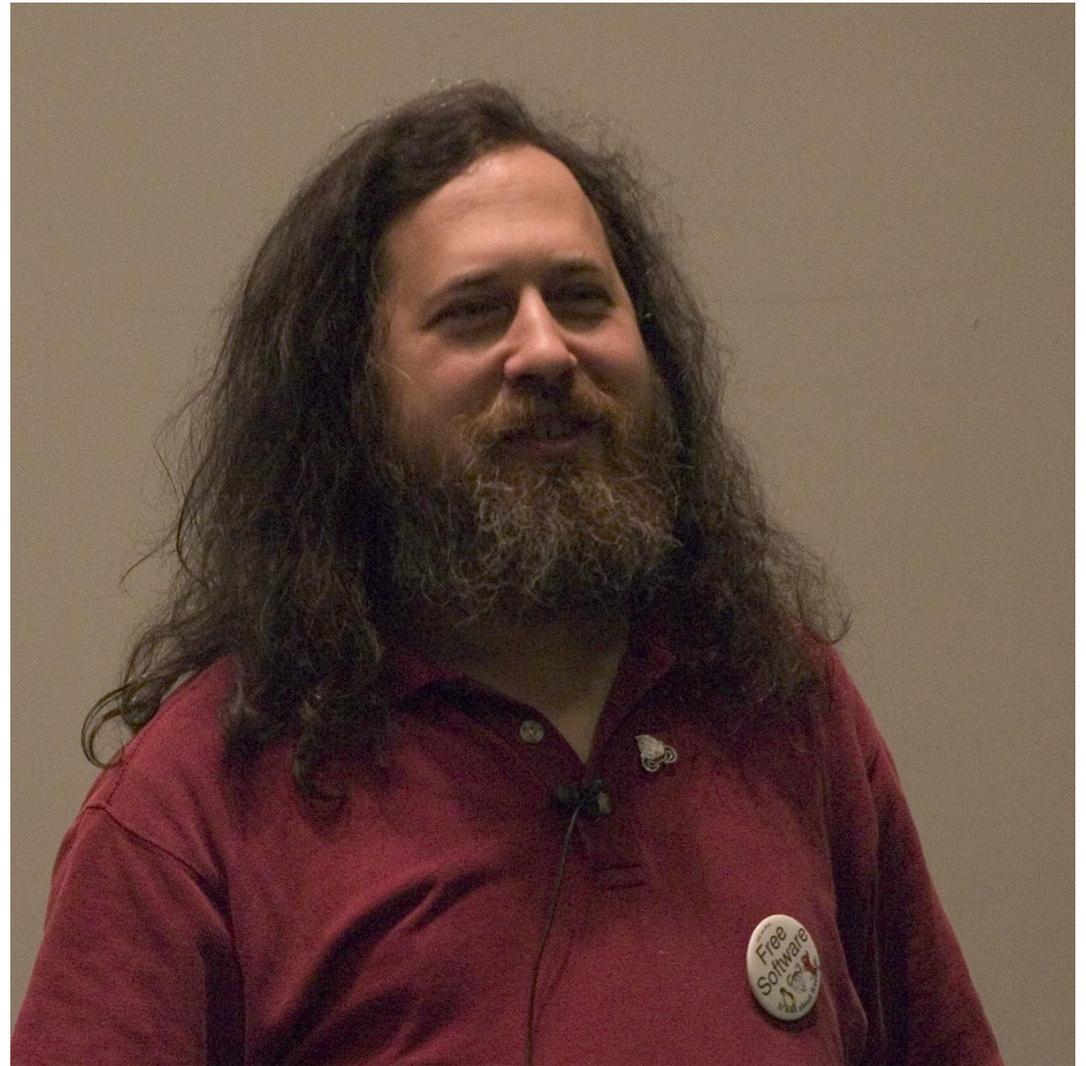
- Doivent être compatibles avec le modèle libre, voir <http://www.opensource.org>
- Environ une centaine de licences
- Problème: compatibilité avec les droits nationaux (ex: droit *moral* en France, n'existe pas dans le droit anglo-saxon)
- Exemples de licences :
 - GPL
 - BSD
 - MIT/X11
 - Mozilla
 - FDL (pour la documentation)

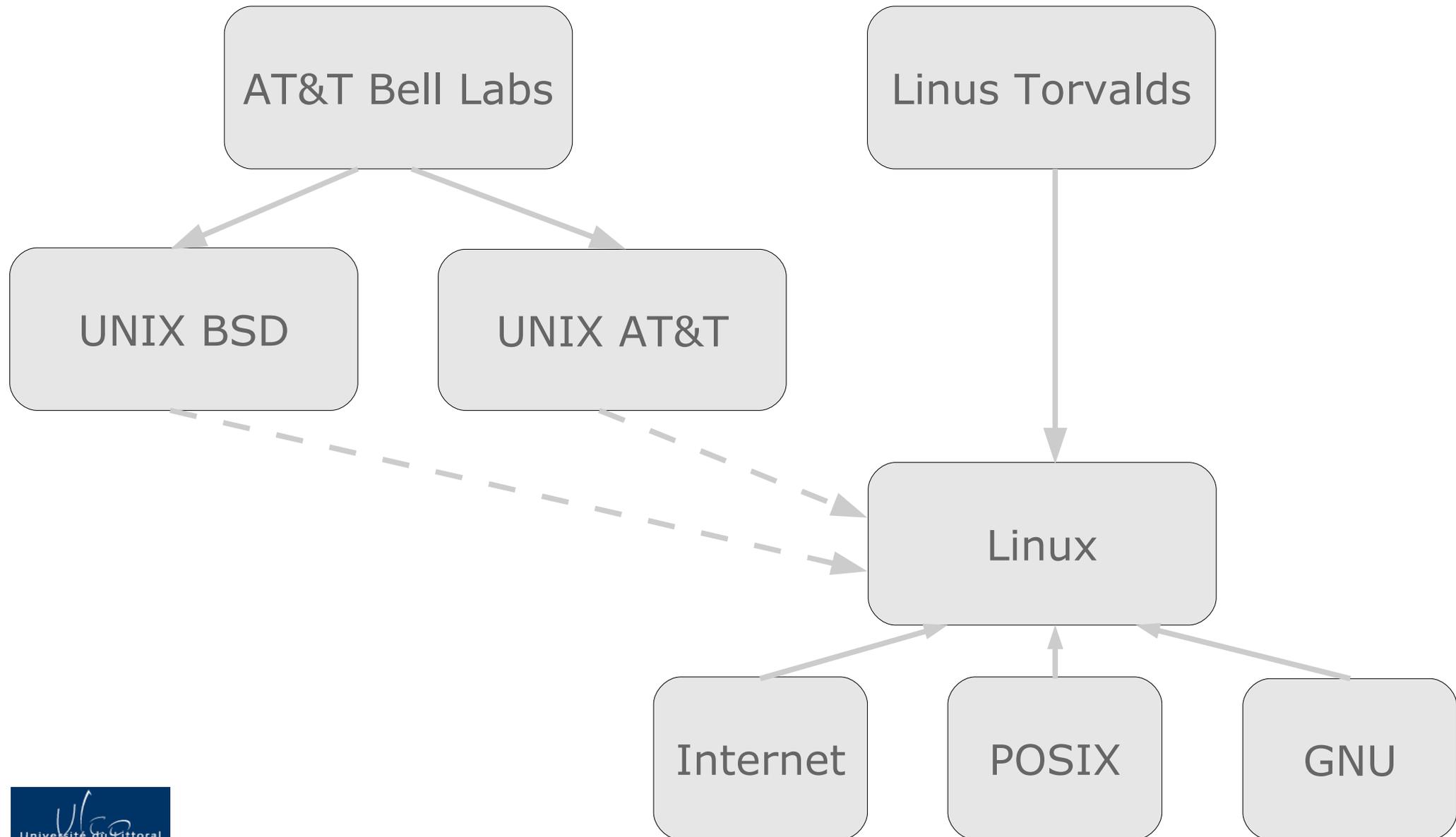
Rappel: les origines de Linux

- Dérivé de la famille UNIX (AT&T, Bell Labs 1970)
- UNIX lui-même dérivé de *MULTICS* (PL/1)
- Volonté de modularité, efficacité, concision
- Influencé par UNIX BSD et AT&T
- A l'origine projet personnel de Linus Torvalds (1991) sur la norme POSIX
- Depuis le début, Linux est un NOYAU et non un système complet => GNU/Linux

- Fortement influencé par le projet GNU (**G**NU is **N**ot **U**nix) de Richard Stallman (MIT années 80)
- Le véritable nom de Linux est donc *GNU/Linux*
- Libre de *toute licence source* par rapport à AT&T
- Internet a également fortement contribué au succès de Linux (plus de connexions en 1990 qu'en 1980 !)
- La personnalité de Linus Torvalds est plus rassurante que celle de R. Stallman!

Les *parents* de Linux !





- GPL = General Public License ou *copyleft*
- La GPL v2 (1991) est la plus répandue (ex: noyau Linux)
- La licence s'applique uniquement en cas de redistribution
- Un code source utilisant du code GPL est du *travail dérivé* et *doit* être publié
- Publication: celui qui reçoit la version binaire peut obtenir le code source
- Pas de lien (link) possible entre du code GPL et du code « propriétaire » !

- Le lien est possible avec la LGPL (Lesser/Library GPL)
- En majeure partie, les bibliothèques système sont diffusées sous LGPL (exemple: glibc)
- Dans le cas d'une application propriétaire il faut donc vérifier qu'aucune bibliothèque « liée » n'est GPL
- Le link *dynamique* n'affranchit pas de la licence
- Solution: utiliser des services (démons) séparés communiquant par IPC ou sockets (exemple: libbluetooth)

- Dans l'espace noyau (pilotes), SEULE la GPL s'applique (en théorie) !
- *You cannot use kernel headers file to create non GPL'd binaries* (Linus)
- En pratique: tolérance si le pilote n'a pas été créé pour Linux (cas du portage)
- Cependant les pilotes binaires posent des soucis techniques (voir présentation sur les pilotes)
- De plus en plus, la propriété intellectuelle reportée dans l'espace utilisateur ou dans un « FPGA »

- Situation fréquente sur des bibliothèques libres supportés commercialement :
 - Une version d'évaluation sous GPL (et non LGPL) => l'utilisateur doit publier son code source sous GPL
 - Une version commerciale si l'utilisateur ne veut pas publier sous GPL
- Exemples:
 - Qt sur <http://www.trolltech.com>
 - Certaines parties de MySQL
 - RTLinux (mauvais exemple...)

- La double licence est possible uniquement si l'exploitant de la licence est également l'auteur du logiciel
- Problème d'ambiguïté vis à vis des contributeurs : ils ont en théorie leur mot à dire en cas de modification du mode de distribution...

- Aspect déontologique: on récupère GRATUITEMENT un système (GNU/Linux) représentant des années.hommes de travail
- Le modèle libre fonctionne grâce aux contributions
- Aspect juridique
 - Incompatibilité avec le droit national (ex: Illiad/Free et la GPL) mais les choses évoluent
 - Site <http://gpl-violations.org>
 - Cependant, la FSF n'est pas un chien de garde ni le BSA...

- Le logiciel libre est associé à des licences, aussi respectables que les licences propriétaires
- La diffusion en toute conformité découle d'une bonne conception de l'architecture (faites vous aider !)
- De nombreuses entreprises (célèbres) ne respectent pas correctement les licences
- La situation évolue favorablement dans la majorité des cas (mais toujours des irréductibles)
- Pas de problème technique sans solution !
- Toujours le problème des brevets logiciels mais pas uniquement sur le logiciel libre