

Le Fonds Européen de Développement Régional et la Région wallonne investissent dans votre avenir



[Jeudis du Libre (Mons) - Jeudi 15 septembre 2011]

La valorisation des logiciels libres en entreprise

Auteur : Dr Ir Robert Viseur



www.cetic.be

Your connection to ICT research

- Robert Viseur
- Ingénieur Civil, Mastère en Management de l'Innovation, Docteur en Sciences Appliquées.
- Spécialisé dans les questions relatives à l'économie des logiciels libres et aux pratiques de co-création, ainsi que dans les technologies de recherche et de traitement de l'information (outils d'indexation, API,...).
- Assistant à la Faculté Polytechnique de l'Université de Mons (www.umons.ac.be).
- Ingénieur de recherche et Conseiller technologique au CETIC (www.cetic.be).
- Plus d'informations: www.robertviseur.be.

Qu'est-ce que le CETIC ?

- Centre d'Excellence en Technologies de l'Information et de la Communication basé à Charleroi (Belgique).
- Trois départements (et types de services):
 - Software & System Engineering : qualité logicielle (fiabilité, sécurité, respect des normes internationales, processus,...).
 - Software & Services Technologies : architectures orientées services et sémantique.
 - Embedded & Communication Systems : prototypage de systèmes embarqués communicants et nouvelles technologies électroniques.

- Partie 1 - Comprendre le logiciel libre: les notions de bases.
- Partie 2 - Comprendre le logiciel libre: les licences.
- Partie 3 - Comprendre le logiciel libre: les modèles d'affaires.

Comprendre le logiciel libre: les notions de bases

Définition de « logiciel libre »

- Logiciel fourni avec 4 libertés (cf. Free Software Foundation, fsf.org):
 - liberté 0: liberté d'exécuter le programme,
 - liberté 1: liberté d'étudier le programme (condition nécessaire: disponibilité du code source),
 - liberté 2: liberté de redistribution de copies (modifiées ou non),
 - liberté 3: liberté de modification (condition nécessaire: disponibilité du code source).
- Remarques :
 - Accès au code source = condition nécessaire mais non suffisante.
 - Pas de référence au prix.

Définition de « logiciel Open Source »

- 10 critères (cf. Open Source Initiative, opensource.org):
 - 1. la liberté de redistribution,
 - 2. la disponibilité du code-source,
 - 3. la possibilité de créer des œuvres dérivés,
 - 4. l'intégrité du code source de l'auteur,
 - 5. la non-discrimination contre des personnes ou des groupes,
 - 6. la non-discrimination contre des champs d'application,
 - 7. la licence s'applique à tout le monde, sans besoin de licence complémentaire,
 - 8. la licence ne doit pas être spécifique à un produit,
 - 9. la licence ne doit pas impacter des logiciels tiers,
 - 10. la licence doit être neutre sur le plan technologique

La « forkabilité » comme conséquence

- Fork: scission d'une partie de la communauté associée à un projet.
- Possibilité inhérente à la définition du logiciel libre.
- Finalement assez rare.
- Causes fréquentes: arrêt de la version libre officielle, problème de gouvernance, divergence d'objectif fonctionnel, propriété d'une marque,...
- Exemples: Sourceforge et GForge, OpenOffice.org et LibreOffice, Inkscape et Sodipodi, Dokeos et Claroline,...

Différence entre logiciel libre et logiciel Open Source

- Différence en terme de définition.
- Différence idéologique:
 - projet politique (éthique) pour la FSF vs...
 - pragmatisme économique et mise en avant du modèle de développement coopératif pour l'OSI.
- En pratique: une licence Open Source est libre et réciproquement.
 - Rares exceptions.
 - Exemple: Apple Public Source License v1.0 et v1.2.
 - Cf. <http://linuxfr.org/news/apple-et-le-libre-lapsl-20-est-une-licence-libre-selon-la-fs>
 - Licences les plus connues: AGPL, GPL, LGPL, MPL, CPL, AL, BSD,...

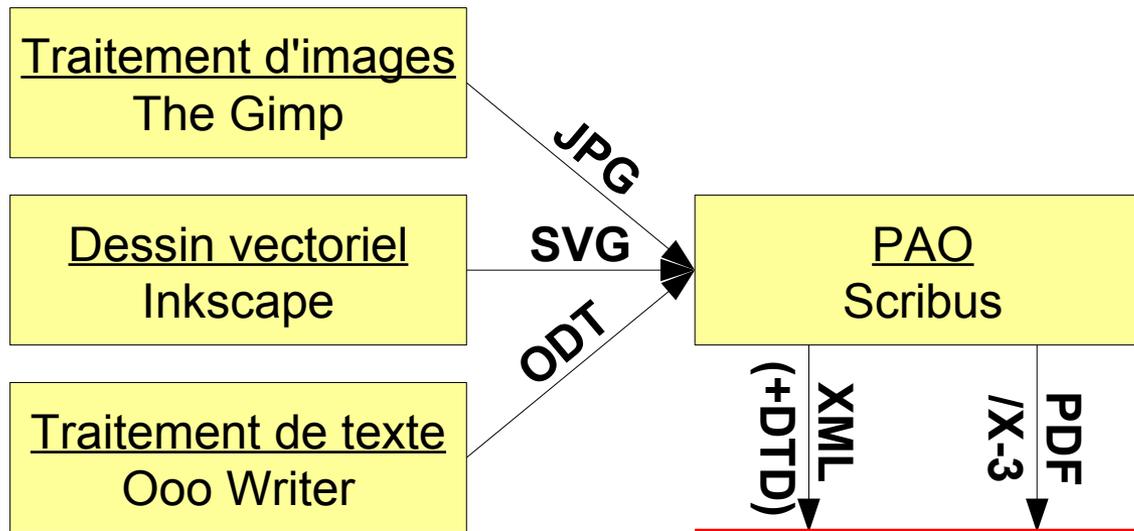
Concept de logiciel ou de licence hybride

- Deux acceptations:
 - 1. logiciel distribué sous licence libre et sous licence commerciale,
 - 2. logiciel distribué sous licence propriétaire "ouverte" (coopération).
- "Notre" logiciel hybride: définition n° 2 (cf. Laure Muselli).
 - Exemples: Sun (cf. Sun Community Source License), Jahia (cf. Sustainable Dual Licensing),...
- Cf. http://fr.wikipedia.org/wiki/Licence_de_logiciel

- Définition belge:
 - « *Par standard, on entend une spécification technique, suffisante pour en développer une implémentation complète, approuvée par un organisme de standardisation indépendant* ».
 - « *Par standard ouvert, on entend un standard qui soit gratuitement disponible sur l'internet et sans restriction juridique quant à sa diffusion et son utilisation* ».
 - Cf. Loi du 10 août 2005 instituant le système d'information Phénix.
- Pour le FEDICT: souhait de plusieurs lecteurs compatibles.

- Remarques:
 - Logiciel libre != Standard ouvert.
 - Bon pour l'intéropérabilité et la concurrence.
 - Long chemin entre la spécification et la ou les implémentations...
 - Norme (si gratuité) > standard ouvert > format ouvert.
 - Usage fréquent des formats et standards ouverts dans les logiciels libres.

- Exemple PAO.



- Cf. RGI (references.modernisation.gouv.fr/rgi-interoperabilite).

Logiciel libre et technologie

- Il est possible de créer du logiciel libre avec du logiciel propriétaire, et réciproquement.
- Exemples: DotNetNuke (MS SQL Server, .Net; licence MIT), Skype for Linux (Qt; freeware),...
- Débat sur la liberté des outils (idée: pas de logiciel libre sans outil libre).
- Attention: possibles conséquences juridiques lors du mélange de codes sources (cf. licences).
- Certaines technologies sont issues du logiciel libre: Perl, PHP, Python,...

Logiciel libre et essaimage

- Application des principes du logiciel libre:
 - Aux bases de données (cf. Open DataBase License; cf. OpenStreetMap).
 - Au matériel (composants électroniques, cartes, machines,...; cf. bibliographie).
 - Aux oeuvres culturelles (cf. Art libre, Creative Commons / CC,...; cf. Jamendo, Culture libre,...).
 - Aux documentations (GNU FDL, Creative Commons,...).
 - Attention aux clauses NC (non commercial) des CC.

- Les entreprises,
- les universités et centres de recherche,
- les utilisateurs et développeurs,
- les communautés,
- les associations,
- les clients
- les administrations.

- Interaction des acteurs au sein d'écosystèmes d'affaires.
- Écosystème d'affaires :
 - « *Ensemble des relations entre acteurs hétérogènes, de secteurs différents, guidés par la promotion d'une ressource et d'une philosophie communes, et de valeurs partagées* »; « *une communauté stratégique d'intérêts, structurée en réseau, autour d'un leader ou d'un groupe qui arrive à imposer ou à transmettre sa conception commerciale, une technologie, un savoir-faire, une norme...* ».
- Cf. Laurence Saglietto.

Coopétition et mutualisation

- Coopétition:
 - Coopération entre concurrents.
 - Exemple: écosystème Eclipse.
 - Les deux plus gros committers sont concurrents: IBM (~22%) et Oracle (~12%).
 - Cf. <http://dash.eclipse.org/dash/commits/web-app/commit-count-loc.php>
- Mutualisation:
 - Développement logiciel entre institutions partenaires (financement et/ou développement).
 - Exemple: CommunesPlone et outils SaaS communaux.
 - Fréquent pour les couches basses (ex.: OW2 et intergiciels).
 - Mutualisation par la demande: coalitions de clients (cf. Elie).

- Différentes stratégies possibles.
- Fonction des degrés de contrôle et de diffusion (variables antagonistes) acceptés par l'entreprise.
- Continuum (valorisation d'une technologie): pas de licence, peu de licences, beaucoup de licences (avec ou sans sélection), Open Source (avec ou sans copyleft).
- Open Source: capitalisation sur une diffusion maximale.
 - Captation des externalités de réseau.
 - Popularisation de la marque, moins de force de vente, partage des coûts de développement,...

- Objectif de diffusion maximale:
 - Freeware,
 - Shareware,
 - SaaS (compte gratuit),
 - freemium,...
- Objectif d'interopérabilité ou de collaboration :
 - standards ouverts,
 - API (en ligne ou plugins),...

Comprendre le logiciel libre: les licences

Quelques bases juridiques (droit d'auteur)

- Un logiciel est protégé par le droit d'auteur.
- Les licences logicielles (contrats) s'appuient notamment sur le droit d'auteur...
- Et fixent les droits et devoirs des parties engagées.
- Point clef: mécanisme clair d'approbation de licence.

Qui est propriétaire ?

- Sous-traitance: vérification des conditions de propriété des codes sources originaux dans le contrat de sous-traitance.
- Contrat de travail (informaticien): présomption de cession des droits patrimoniaux à l'employeur.
- Rappel: droits patrimoniaux = droits de reproduction, de distribution et de communication au public.

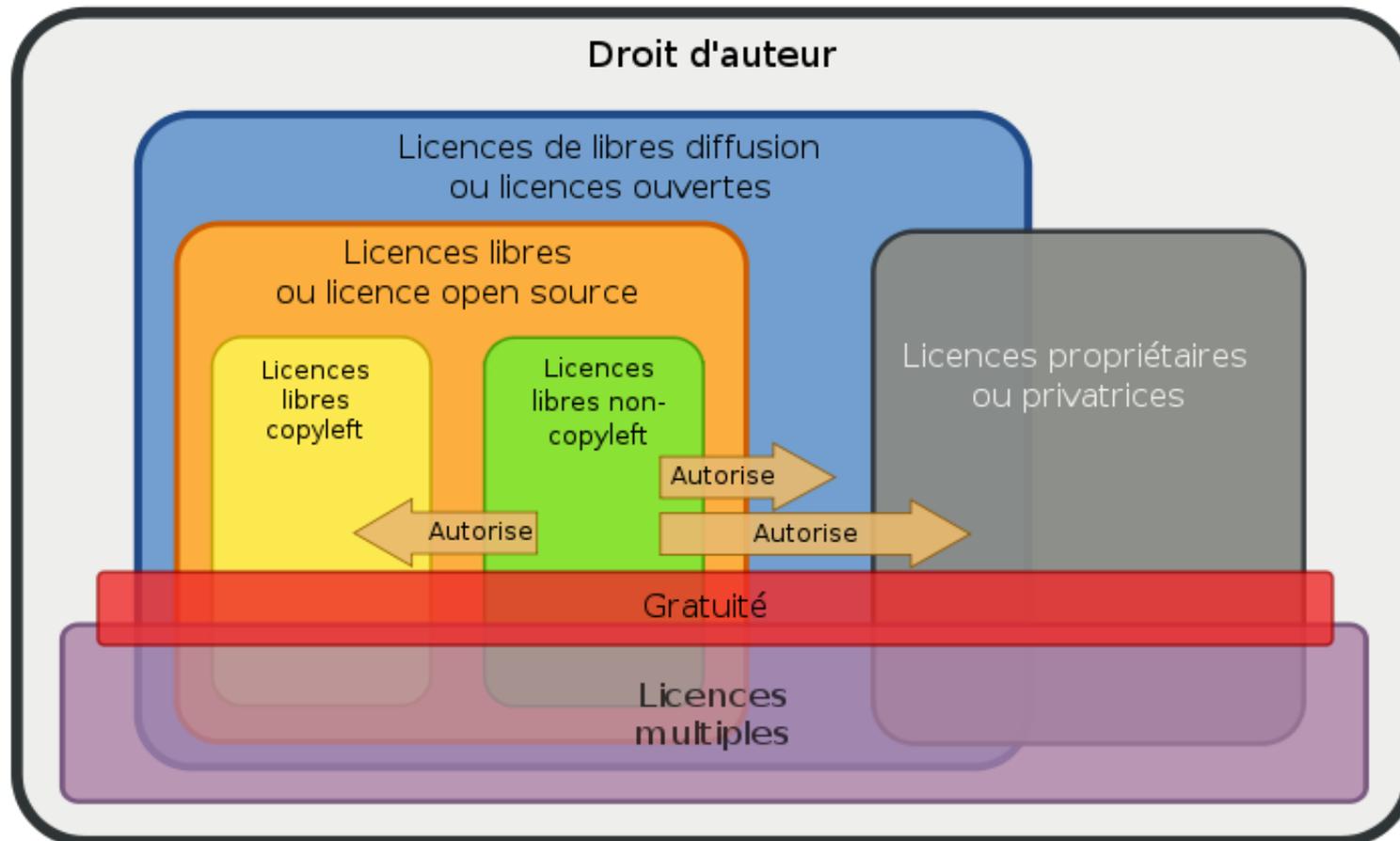
- IP : droit d'auteur, marques, brevets,...
- Protection de base: droit d'auteur (ou copyright).
- Et donc: les licences logicielles (contrats).
- Autres:
 - droit d'auteur (documentations) et, donc, licences (ex.: CC0, CC-BY-SA, GFDL,...),
 - droit des marques (nom, logo),
 - brevets logiciels (USA, Japon,...; pas en Europe sauf cas de jurisprudence),
 - schémas électroniques (cf. bibliographie),
 - bases de données (ex.: ODBL).

Types de licences libres (1/2)

- Deux grandes familles:
 - académiques (ou permissives),
 - copylefts (ou gauches d'auteur) faibles ou fortes.
- En pratique:
 - Une licence permissive permet le changement de licence (ex.: Paint.Net passé de licence MIT à freeware).
 - Une licence copyleft impose la conservation de la licence.
 - Remarque: la possession de tous les droits patrimoniaux (ou, à défaut, un accord de tous les contributeurs) sur le code source permet cependant d'apposer une autre licence, même si le code source est sous licence copyleft.
 - Une licence copyleft forte impose l'extension de la licence à toute oeuvre dérivée.
 - Une licence copyleft faible considère une application utilisant le logiciel comme n'étant pas une oeuvre dérivée (définition approximative, cf. infra).
- Si plusieurs licences: choix de la licence par l'utilisateur.

Types de licences libres (2/2)

- Classification des licences :



Principales licences

(...)

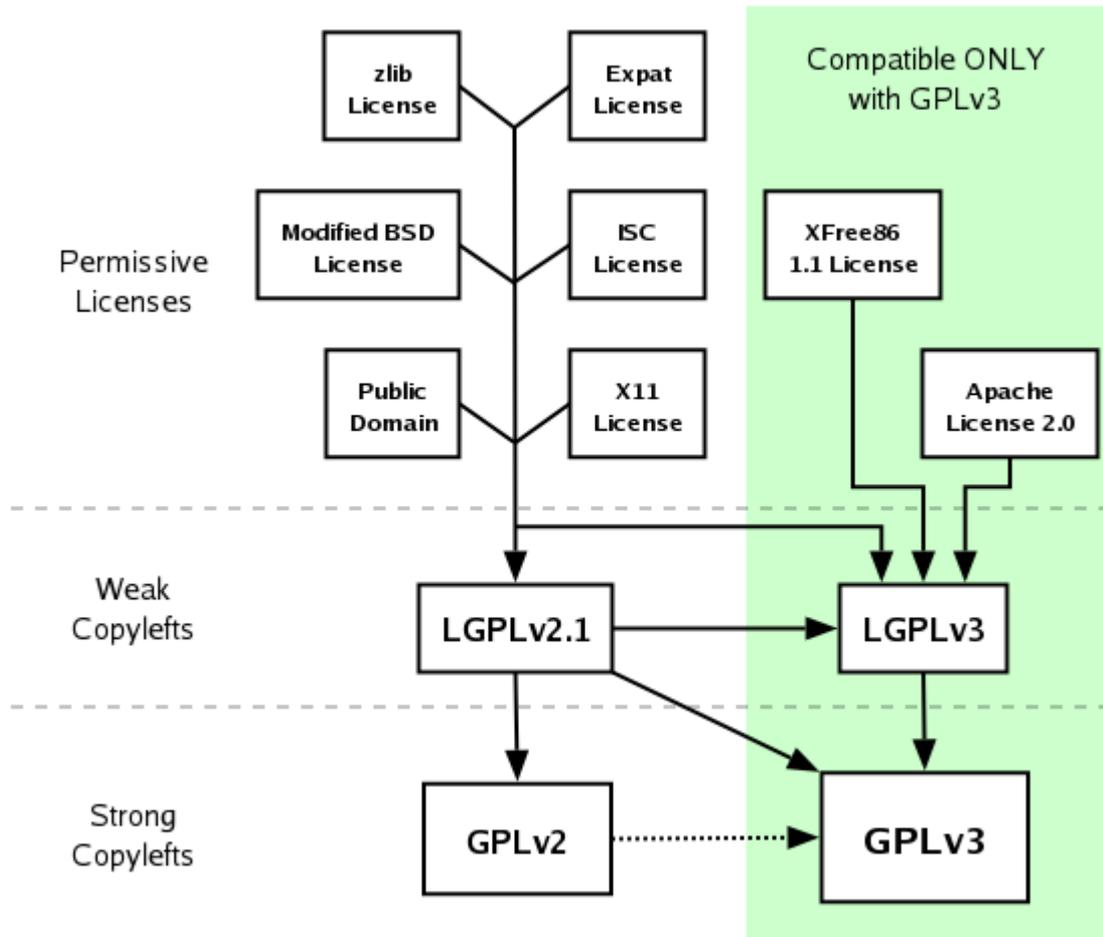
Compatibilité entre licences libres

(1/2)

- Nombreuses licences libres et Open Source.
- Parfois: plusieurs versions d'une même licence (ex.: GPL v2 et GPL v3).
- Incompatibilités entre licences.
 - Complexe.
 - Exemples:
 - AL v2 <nok> GPL v2 mais AL v2 -ok> GPL v3.
 - GPL v2 <nok> GPL v3 mais GPL v2+ -ok> GPL v3.
 - ...
- Exemple de problème : KDE (cf. http://techbase.kde.org/Projects/KDE_Relicensing).

Compatibilité entre licences libres (2/2)

Cf. <http://open-source.smile.fr/Copyright-et-licences>



Compatibilité entre licences et places de marché

- Développement des place de marché logicielles.
- Canal de distribution dominant pour les OS mobiles.
 - Cf. Android Market, Appstore, Windows Marketplace,...
- Contraintes à la diffusion de logiciels libres sur Appstore et Windows Marketplace:
 - Pas explicite pour l'Appstore (réduction de libertés via EULA),
 - Explicite pour le Windows Marketplace (cf. Excluded License)
 - Rejet des licences copyleft.
 - Exceptions: CDDL, MPL, EPL, CPL et MS-RL.
 - Rejet mis en emphase pour GPL v3 et LGPL v3.
- Clauses fréquemment changeantes.

Attention aux fausses licences libres et Open Source !

- Exemple: SuperForm Community License (cas réel, nom fictif).
 - Présentée comme Open Source et de type EUPL.
 - EUPL? Licence copyleft multilingue de l'EU adaptée au cadre légal européen et incluant une importante clause de compatibilité avec d'autres licences libres.
 - Ici:
 - reprise du texte de base mais...
 - suppression des clauses copyleft et de compatibilité et...
 - ajout d'une clause de gratuité.
- > licence non libre, non Open Source!

- Prévoir des mécanismes de décision pour:
 - les publications sous licence libre,
 - le choix de la licence,
 - l'utilisation de codes libres (édition),
 - la surveillance des codes produits en interne.
- But ? Éviter les surprises à la publication (ex. : logiciel électoral belge, cf. <http://shrl.be/000082>)...

Analyse juridique de code source

(...)

Comprendre le logiciel libre: les modèles d'affaires

- Objectifs de la libération?
 - Réduction des coûts?
 - Changement des règles sur un marché?
 - Création de chiffre d'affaires (de manière directe ou indirecte)?
 - ...

- Modèles d'usage: plus de licence à payer, d'où économies d'échelle si nombreuses installations (ex.: Google et "son" Linux)
- Développement en commun de composants logiciels non différenciants (ex.: Apple et Darwin, OW2 et intergiciels,...).
- Cas particuliers: mutualisation par la demande, coopération.
- Accélération de la diffusion de logiciels de base (ex.: pilotes d'imprimante HP).

Changement de règles (1/2)

- En cas de marché très stable (ex.: oligopole concentré, monopole): dynamitage du marché.
 - Publication d'un logiciel ou d'un standard concurrent d'une technologie commerciale dominante (lock-in).
 - Principe: tuer le "cash cow" du concurrent.
 - But: affaiblir le concurrent et...
 - Faciliter une attaque sur un autre marché ou...
 - Simplement se faire (ou garder) une place (captation d'externalités).

Changement de règles (2/2)

- Exemples:
 - Sun Microsystems et OpenOffice.org (contre Microsoft Office = \$\$\$).
 - Contexte: concurrence sur les outils de développement Java et J2EE vs C# et .Net).
 - SAP et MaxDB (contre Oracle Database).
 - Contexte: concurrence sur les outils PGI.
 - OpenERP (contre SAP, Oracle,...),
 - Jabber et standard XMPP (vs Messenger, ICQ,...).

- Activités classiques de prestations de services.
 - Migrations de logiciels propriétaires,
 - Intégration de logiciels libres,
 - Formation à l'usage de logiciels libres,
 - Développement sur mesure,
 - ...

- Hébergement classique ou SaaS (*Software as a Service*).
- Offre logicielle en SaaS.
 - Exemples:
 - OpenERP et OpenERP Online,
 - ERP5 et TIOLive (www.tiolive.com).
 - Concept de SaaS libre ou ouvert.
 - Cf. TIO Libre et TIO Openess (www.tiolibre.com).

- Différentes formes d'activités d'édition:
 - Packaging de logiciels existants avec garantie de bon fonctionnement (abonnement pour mises à jour; cf. Red Hat).
 - Développement d'un logiciel (produit "vedette") et:
 - Prestation de services et/ou...
 - Hébergement en SaaS et/ou
 - Double licence.

Double licence (licence chronodégradable)

- Publication sous une licence propriétaire à durée limitée.
- Publication sous licence libre à l'expiration de la licence propriétaire.
- Exemple: AFPL Ghostscript (non libre) et GNU Ghostscript (jusque 2006).

Double licence (sans différenciation technique)

- Édition sous licence commerciale (propriétaire) et sous licence libre.
- La licence libre est généralement la GPL (copyleft fort).
 - Ex.: Trolltech Qt (avant le rachat par Nokia).
 - Contexte: souvent pour du logiciel servant à la création d'oeuvres dérivées.
 - La GPL impose l'achat d'une licence commerciale pour l'édition sous licence autre que GPL.
 - Publier en GPL ou payer.

Double licence (avec différenciation technique)

- Edition sous licence commerciale (propriétaire) et sous licence libre.
- La licence libre est généralement copyleft.
 - L'utilisateur reste soumis au devoir de contribution.
 - Ex.: Compiere Community sous GPL; Compiere Standard, Professional et Enterprise sous licence commerciale.
- L'éditeur rajoute des fonctionnalités complémentaires (issues ou non de tierces parties) dans la version propriétaire.

Double licence (précautions nécessaires)

- Nécessité de disposer des droits patrimoniaux sur le code source via:
 - la réimplémentation des contributions (ex.: MySQL),
 - la signature d'un accord de contributeur (ex.: Sun Contributor Agreement / SCA),
 - un incitant financier au partage de droits (ex.: Exo).
- Ou utiliser une licence permissive ou copyleft faible (ex.: MPL pour Netscape / Mozilla).

- Architecture noyau extension fréquente pour les logiciels libres (ex.: Drupal).
- Variante: Open Core.
 - Modèle noyau extension avec noyau générique et extensions spécialisées et payantes.
 - Vente parfois organisée au sein de places de marchés (ex.: Magento Connect).
 - Parfois: revenus complémentaires sous la forme de commissions sur les ventes.
 - Principe de l'effet Halo.

- Vente de produits dérivés:
 - extensions par prolongation de cycle de vie (ex.: de jeux pour ID Software),
 - livres (ex.: O'Reilly),
 - merchandising,
 - ...
- Publicité (ex.: Firefox et Google AdSense).

Combinaisons possibles

- Exemple: boutique en ligne *Magento* (*www.magentocommerce.com*).
 - *Magento Community Edition* sous licence OSL (copyleft fort).
 - *Magento Professional* et *Enterprise* sous licences propriétaires payantes (avec discrimination fonctionnelle).
 - Hébergement SaaS avec *Magento Go*.
 - Extensions gratuites ou payantes sur *Magento Connect*.
 - Services de consultance et de formation.

- La valorisation des logiciels libres peut se faire de plusieurs manières (réduction des coûts, attaque d'un monopole, génération de revenus,...).
- L'écosystème du logiciel libre accorde une place importante à la collaboration.
- Les règles de collaboration sont (partiellement) réglées par les licences de logiciels libres.
- Ces licences sont (presque) toujours à la fois libres et Open Source.

- Les licences de logiciels libres peuvent avoir un impact sur les modalités de valorisation des logiciels édités ou sur vos actifs immatériels (ex. : marques, brevets,...).
- Les licences de logiciels libres présentent des incompatibilités entre elles ou avec les EULA d'autres éditeurs (ex. : Microsoft Marketplace).
- Les licences de logiciels libres sont des textes juridiques parfois complexes mais connus, stables et largement commentés (pour les plus diffusées).

- En cas de choix d'une licence (édition), la licence doit être en phase avec vos choix de valorisation (modèle d'affaires).
- En cas d'utilisation de composants sous licence libre dans des développements courants, la mise en place d'actions en interne (sensibilisation des développeurs et chefs de projets, formations, audits de codes sources, processus de validation,...) est recommandable.
- La publication sous licence libre est un des moyens (d'autres existent) pour accroître la collaboration autour d'un logiciel.

Merci pour votre attention.
Des questions ?

- Benoit Demil et Xavier Lecocq (2003). "Comment exploiter brevets et marques ?". L'Expansion Management Review, juin 2003, p88-95.
- François Elie (2009). Economie du logiciel libre. Eyrolles, 5 février 2009.
- Etienne Montero et al. (2005). Les logiciels libres face au droit. Cahier du CRID n°25. Bruylant.
 - Laurence Saglietto (2007). De la place de marché électronique à l'écosystème d'affaires: le cas des systèmes de réservations aériennes et touristiques. Systèmes d'Information et Management. Editions ESKA. N° 3 vol. 12, p77-93.
- Robert Viseur (2011). Introduction à l'Open Hardware. Rencontres Mondiales du Logiciel Libre 2011, Strasbourg, 11 juillet 2011.
- Robert Viseur (2011). Cartographie des marchés Open Source belges et français. Rencontres Mondiales du Logiciel Libre 2011, Strasbourg, 11 juillet 2011. Cf. <http://robertviseur.be/news-20110721.php>
- Robert Viseur (2010). Schémas de valorisation Open Source. Rencontres Mondiales du Logiciel Libre 2010, Bordeaux, 09 juillet 2010. Cf. <http://www.robertviseur.be/news-20100710.php>
- Robert Viseur (2010). "La co-création ou comment innover avec le client?", Centre d'entreprise Héraclès, 26 mai 2010.
- Robert Viseur (2010). Logiciels libres et concurrence. "Logiciels libres, une fenêtre ouverte sur l'avenir?", SPF Economie, 12 mars 2010. Cf. <http://www.robertviseur.be/news-20100406.php>
- Robert Viseur (2009). "Quand valoriser en Open Source ?", Open The Source I, Charleroi, le 25 novembre 2009. Cf. <http://www.robertviseur.be/news-20091217.php>

- Dr Ir Robert Viseur.
- Email : robert.viseur@cetic.be
- Phone : 0032 (0) 479 66 08 76

Cette présentation est diffusée sous licence « CC-BY ».