
Droit des logiciels

...ou du moins un petit aperçu !

François Pellegrini
Professeur, Université de Bordeaux
francois.pellegrini@u-bordeaux.fr

Ce document est copiable et distribuable librement et gratuitement à la condition expresse que son contenu ne soit modifié en aucune façon, et en particulier que le nom de son auteur et de son institution d'origine continuent à y figurer, de même que le présent texte.

Droit et informatique

- Droit et informatique sont des disciplines de même nature
 - Elle manipulent le « code »
 - « Law is Code », tout autant que « Code is Law »
 - Elles consistent à construire, à partir de quelques éléments de base, les règles de fonctionnement d'univers complets
 - Le problème, c'est que c'est l'univers dans lequel on vit !
 - Après le « monde de Kafka », le « monde de Matrix »
- Intérêt d'une compréhension mutuelle entre informaticiens et juristes

Quelques notions d'économie des biens immatériels...

Copier n'est pas voler...

Économie des biens immatériels

- L'économie des biens immatériels diffère fondamentalement de l'économie matérielle
 - Biens non rivaux
 - Le coût marginal de la copie est nul
 - Une œuvre numérique peut être distribuée gratuitement dès le moment où son développement a été financé
 - Les effets de réseau sont considérables
 - La valeur d'un produit augmente avec le nombre de personnes qui l'utilisent
 - Très grande volatilité du marché
 - Obsolescence très rapide

Un logiciel non utilisé est un logiciel qui meurt

Histoire du logiciel

Et le logiciel fut !

Apparition du logiciel (1)

- Les premiers calculateurs ne possédaient pas de logiciel
 - Programmés « à la main »
- Le logiciel est apparu avec l'idée que le programme pouvait être stocké dans la même mémoire que celle utilisée pour les données
 - Architecture dite « de von Neumann »
 - Le logiciel est une donnée (presque) comme une autre
 - Possibilité de programmes auto-modifiables
 - Méta-programmation

Apparition du droit du logiciel (1)

- Tant que les logiciels étaient spécifiques à un ordinateur donné, la question de leur statut ne se posait pas
 - Les logiciels étaient fournis « gratuitement » aux utilisateurs, en tant que « fournitures annexes »
 - Au même titre que les manuels d'utilisation

Apparition du droit du logiciel (2)

- Les fournisseurs encourageaient leurs clients à modifier les logiciels et à partager les modifications
- Soutien à la création de « clubs d'utilisateurs » servant à l'échange de ces améliorations
 - Mutualisation et donc réduction des coûts de maintenance logicielle, offrant un avantage compétitif sur les concurrents
 - Les principes du logiciel libre avant l'heure !

Apparition du droit du logiciel (3)

- La question du statut juridique du logiciel s'est posée lorsque sont apparus les premiers ordinateurs compatibles avec les grands systèmes IBM, à la fin des années 1960
 - Les fournisseurs de matériels compatibles utilisaient les logiciels d'IBM sans en payer le prix
- IBM décida donc de facturer séparément logiciel et matériel
 - Politique de « dégroupage » (« unbundling »)
 - Mais comment caractériser juridiquement le logiciel ?

Quel statut pour le logiciel ? (1)

- Trois voies offertes au législateur :
 - Droit des brevets
 - Jugé inadapté de par l'objet à réguler et la lourdeur des mécanismes de dépôt et d'entretien
 - Droit « sui generis »
 - Taillé sur mesure
 - Durée de mise en œuvre des conventions internationales
 - Droit d'auteur
 - Processus de création similaire entre logiciels et œuvres littéraires
 - Existence de la Convention de Berne (1886)
 - Protection internationale automatique et immédiate

Quel statut pour le logiciel ? (2)

- C'est le droit d'auteur qui fut choisi comme véhicule du droit du logiciel
 - 1980 aux États-Unis (modification du Copyright Act)
 - 1985 en France (loi du 3 juillet 1985)
 - 1991 au sein de l'Union européenne (directive 91/250/CE)
 - 1994 parmi les membres de l'OMC (accords ADPIC)
 - 1996 au niveau mondial (traité WCT de l'OMPI)

Quel statut pour les langages ? (1)

- Un langage n'est pas une œuvre
 - Il permet d'écrire des œuvres
 - Il est d'un niveau d'abstraction supérieur
 - Pas de revendication possible sous le régime du droit d'auteur
- Un langage informatique est aussi une langue de communication humaine
 - Permet l'échange d'informations entre humains

Quel statut pour les langages ? (2)

- Les langages informatiques ne peuvent donc être appropriés sans porter préjudice à des droits plus élevés
- Un format de fichiers définit la grammaire d'un langage
 - Accorder des brevets sur des formats de fichiers reviendrait donc à pouvoir monopoliser un langage
 - Donc aussi par extension des éléments de langues humaines

Le droit d'auteur

...ou, plus exactement, un bref aperçu
d'icelui

Droit des biens immatériels

- Définis en France dans le « Code de la Propriété Intellectuelle » (CPI)
 - Terme « impropre », car c'est un oxymore
- Organisation hiérarchique du Code :
 - Code de la propriété littéraire et artistique
 - Droits d'auteur
 - Droits « voisins des droits de l'auteur »
 - Droits des interprètes
 - Code de la propriété industrielle
 - Droit des brevets
 - Droit des marques

Justification du droit d'auteur (1)

- Encourager globalement la création en garantissant aux créateurs un monopole temporaire sur l'exploitation de leurs créations
- Au terme de ce monopole, les créations gagnent le domaine public, et peuvent ainsi bénéficier à tous
 - Une œuvre ne tombe pas dans le domaine public, elle s'y élève !

Justification du droit d'auteur (2)

- Le monopole temporaire concédé au titre du droit d'auteur représente un compromis entre l'intérêt des auteurs et l'intérêt du public

« Le livre, comme livre, appartient à l'auteur, mais comme pensée, il appartient [...] au genre humain. [...] Si l'un des deux droits, le droit de l'écrivain et le droit de l'esprit humain, devait être sacrifié, ce serait, certes, le droit de l'écrivain, car l'intérêt public est notre préoccupation unique, et tous [...] doivent passer avant nous. »

Victor Hugo,
Discours d'ouverture du congrès littéraire, 1878

Structuration du droit d'auteur

- Organisation hiérarchique des droits d'auteur :
 - Droits patrimoniaux
 - Attachés à l'œuvre
 - Droits extra-patrimoniaux, aussi appelés droits moraux
 - Attachés à la personne de l'auteur (ou à ses descendants)
 - N'existent pas dans le système du copyright

Œuvre (1)

- L'œuvre est une création de forme
 - C'est la forme qui sera protégée, et non les idées et les concepts
- « Les idées sont de libre parcours » !

Œuvre (2)

- Liste non limitative des œuvres susceptibles de recevoir la protection du droit d'auteur :
 - « Notamment » (L.112-2 CPI) : les livres, brochures et autres écrits littéraires, artistiques et scientifiques ; les conférences, allocutions, sermons, plaidoiries et autres oeuvres de même nature ; les oeuvres dramatiques ou dramatico-musicales ; les oeuvres chorégraphiques, les numéros et tours de cirque, les pantomimes, etc...

Critère de protection

- Notion d'« originalité », reflétant la « personnalité de l'auteur »
 - Reformulé sous le terme d'« apport intellectuel » lorsque sera abordée la question des œuvres logicielles
- Une œuvre non originale ne sera pas susceptible de protection :
 - Photographie conforme d'un tableau ancien
 - Mais une photo floue, si !
 - Photos « non artistiques » de plats de cuisine
 - Photo prise par un animal non-humain

Critères non pertinents (1)

- La « nouveauté »
 - « Encore une histoire de mousquetaires ?! »
 - Des navigateurs web différents sont bien chacun des œuvres originales

Critères non pertinents (2)

- Le « mérite »
 - Qui se souvient encore des peintres « officiels » de la fin du XIXe siècle, par rapport aux « refusés » ?
 - Aux États-Unis, les droits concédés le sont « To promote the Progress of Science and useful Arts »
 - Certaines œuvres considérées comme « indécentes » ont été jugées indignes de recevoir la protection par le copyright
 - Argument utilisé par ceux qui copient du porno !

Créations échappant à la protection

- Une création de forme ne reflétant pas la personnalité de son auteur ne sera pas éligible à la protection par le droit d'auteur
 - Tables mathématiques, annuaires des marées, etc.
- Cependant, leur présentation graphique pourra être éligible si elle reflète la personnalité de son auteur (logos, décorations, etc.)

Automaticité de la protection (1)

- La protection est réputée acquise dès la conception de l'œuvre
 - L.111-1 CPI : « *L'auteur d'une œuvre de l'esprit jouit sur cette œuvre, du seul fait de sa création, d'un droit de propriété incorporelle exclusif et opposable à tous* »
 - L.111-2 CPI : « *L'œuvre est réputée créée, indépendamment de toute divulgation publique, du seul fait de la réalisation, même inachevée, de la conception de l'auteur* »

Automaticité de la protection (2)

- Aucune formalité d'enregistrement n'est nécessaire
- Il est prudent de se pré-constituer des preuves de paternité et d'antériorité :
 - Constat d'huissier
 - Dépôt chez un notaire
 - Dépôt auprès d'associations spécialisées
 - SGDL, APP, etc.
 - Courrier envoyé à soi-même
 - etc.

Droits patrimoniaux

- Matérialisation du droit qu'a l'auteur de tirer profit de l'utilisation de son œuvre
 - Protection économique de l'œuvre
- Distincts de la possession physique de l'œuvre
 - On n'achète que la « matière » du tableau, pas ce qu'il représente
 - La cession des droits de reproduction doit être explicite
- Les droits patrimoniaux sont :
 - Cessibles
 - Temporaires

Durée des droits patrimoniaux

- La durée des droits patrimoniaux dépend des pays et des situations
 - Mais uniformisation mondiale sous la pression de l'industrie du divertissement
- Actuellement, au sein de l'Union Européenne, cette durée est de 70 ans après le décès de l'auteur
 - Délai peu compatible avec la préservation du patrimoine logiciel

Droits extra-patrimoniaux

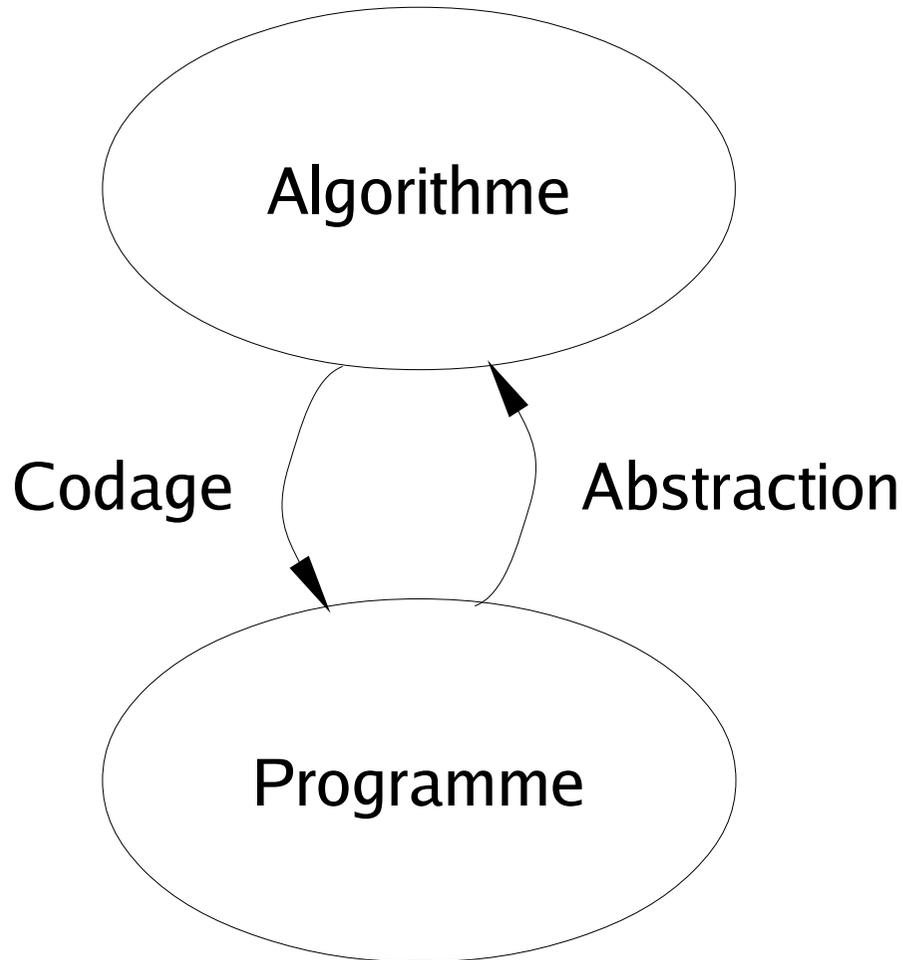
- Protègent tant l'œuvre que l'acte de création
- Trois droits distincts :
 - Droit à la paternité
 - Les contrats « de nègre » sont illégaux et jugés comme tels
 - Droit au respect de l'œuvre
 - Droit spécifique pour le nom de l'œuvre
 - Droit de divulgation
- Les droits extra-patrimoniaux sont :
 - Inaliénables
 - Perpétuels

« Exceptions » aux droits de l'auteur

- Définies par la loi
 - Ce ne sont donc pas des exceptions individuelles mais un droit !
- Liste exhaustive incluant :
 - Copie privée à l'usage du copiste
 - Représentation dans le cercle de famille
 - Citation
 - Parodie
 - etc.

Le droit d'auteur adapté au logiciel

Nature du logiciel



- Les algorithmes sont :
 - Des idées
 - Des mathématiques
- Les programmes sont :
 - Des œuvres de l'esprit
 - Du discours
 - Humain ↔ humain
 - Humain → ordinateur
 - Des processus, lorsqu'ils sont exécutés

- **Similaire au processus de création littéraire**

Droit d'auteur adapté

- Par son rattachement au droit d'auteur, le logiciel est assimilé à une œuvre de l'esprit
 - En France, loi du 3 juillet 1985
 - Directive européenne 91/250/CE (1991)
 - Article 10 des accords ADPIC (1994)
 - Article 4 du traité OMPI WCT (1996)
- Le logiciel est cependant aussi un produit substituable voué à rendre un service
 - Adaptation du droit d'auteur
 - Nous parlerons donc de « Droit d'auteur adapté »
 - Question de la garantie

Adaptation des droits patrimoniaux (1)

- Ajout de nouvelles exceptions
 - « Actes nécessaires pour permettre l'utilisation du logiciel »
 - « Observation du fonctionnement »
 - En fait, simple rappel de la loi
 - Copie de sauvegarde
 - Si aucun autre moyen fourni par l'éditeur
 - La décompilation
 - Interdite sauf « à fin d'interopérabilité »
- Suppression de l'exception de copie privée
 - Nécessite une licence pour chaque exemplaire du logiciel

Adaptation des droits patrimoniaux (2)

- Pour les auteurs salariés ou fonctionnaires, transfert automatique de la titularité des droits patrimoniaux à l'employeur
 - C'est l'employeur qui décide de la vie de l'œuvre :
 - Divulcation, choix de la licence, etc.
 - Apparition d'un statut d'« auteur prolétaire »
 - Déjà entamé avec la notion d'« œuvre collective »
 - Ne concerne que les logiciels réalisés dans le cadre professionnel, pendant le temps de travail !

Adaptation des droits patrimoniaux (3)

- Les auteurs non salariés restent titulaires des droits patrimoniaux sur leurs œuvres logicielles
 - Nécessité de transférer les droits des sous-traitants
 - Problème des stagiaires non rémunérés

Adaptation des droits extra-patrimoniaux

- Amoindrissement des droits extra-patrimoniaux
 - L'auteur salarié ou fonctionnaire ne peut s'opposer à la modification de l'œuvre
 - Reste le droit au nom...
 - Plus courant dans le monde du jeu vidéo que de la comptabilité

Compilation, décompilation et interopérabilité

Faire et défaire,
c'est toujours travailler !

Langages de bas et haut niveau (1)

- Les ordinateurs ne comprennent que le « langage machine »
 - Instructions très rudimentaires
 - Spécifique à un processeur donné
 - Et à un système d'exploitation donné
 - Code très peu expressif
 - Risques de bogues élevé
 - Proportionnels au nombre de lignes que l'on écrit

Langages de bas et haut niveau (2)

- Besoin d'écrire des programmes dans des langages plus expressifs, dits « de plus haut niveau »
 - Meilleure compréhension du code
 - Indépendance vis-à-vis du type de processeur
 - Meilleures maintenabilité et portabilité
 - Coût économique bien inférieur

Compilation et décompilation (1)

- La compilation consiste à traduire un programme écrit dans un langage de haut niveau en un programme fonctionnellement équivalent écrit dans un langage de bas niveau, susceptible d'être exécuté par un ordinateur
 - Le « code source » est le programme écrit en langage de haut niveau que l'on veut traduire
 - La « forme préférée » d'écriture d'un programme
 - Le « code objet » est le programme résultant écrit dans un langage de bas niveau
 - Définit le « programme exécutable »

Compilation et décompilation (2)

- On appelle « décompilation » l'action inverse de la compilation, permettant d'exprimer dans un langage de haut niveau un programme originellement écrit dans un langage de bas niveau
 - Bien plus difficile à mettre en œuvre
 - Informations structurelles de haut niveau « diluées » dans le code objet

Traduction automatique

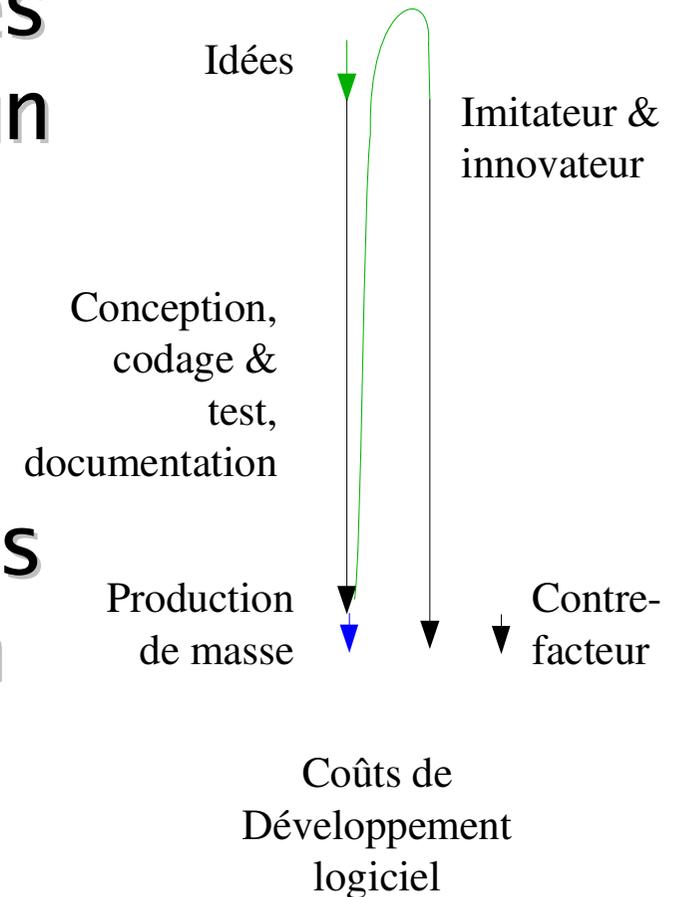
- La traduction du code source en code objet se faisant de façon automatique, le code objet est une œuvre intégralement dérivée du code source, sans aucun apport original d'un autre auteur
 - Le créateur d'un outil n'a aucun droit sur les créations réalisées au moyen de cet outil
 - Pensez aux dictionnaires, pinceaux, etc...
 - Cas particulier : inclusion de fragments de code du compilateur dans le programme exécutable
 - Statut des intelligences artificielles
 - « Valladolid 2.0 » !

Interdiction de la décompilation

- Compiler et décompiler créent des œuvres dérivées de l'œuvre originale
- Ces copies ne peuvent donc être exploitées qu'avec l'accord de l'ayant droit des œuvres originales
- La décompilation est donc interdite, sauf permission explicitement donnée par l'ayant droit

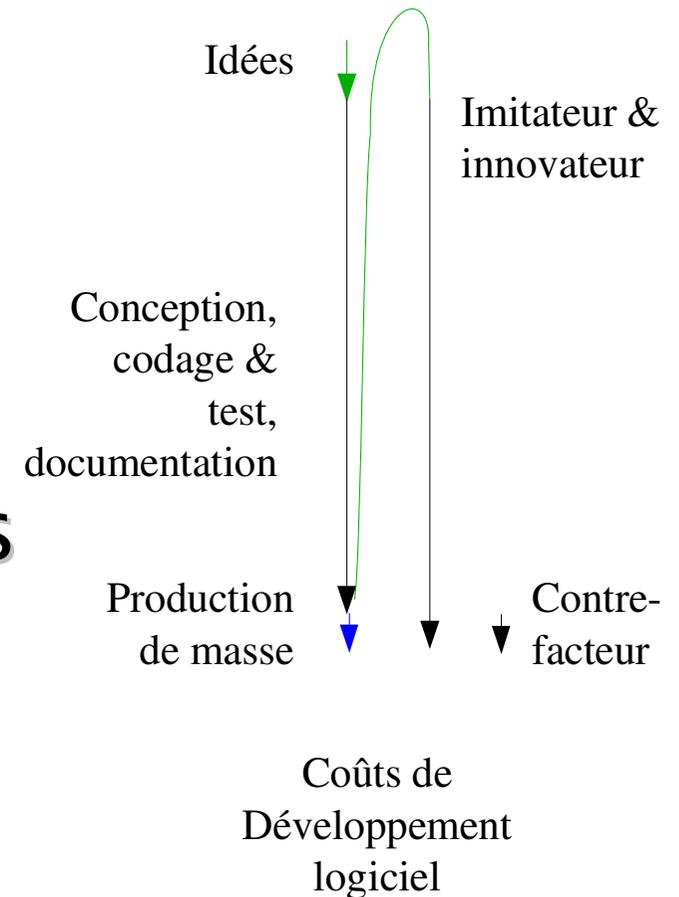
Modèle économique de la création logicielle (1)

- L'observation des fonctionnalités d'un logiciel existant permet à un nouvel arrivant de réaliser un logiciel concurrent
- Il n'y a concurrence libre et non faussée que s'il paye les mêmes coûts d'entrée sur le marché en re-codant son logiciel
 - Il ne peut donc réutiliser le code existant, par décompilation et/ou copie servile



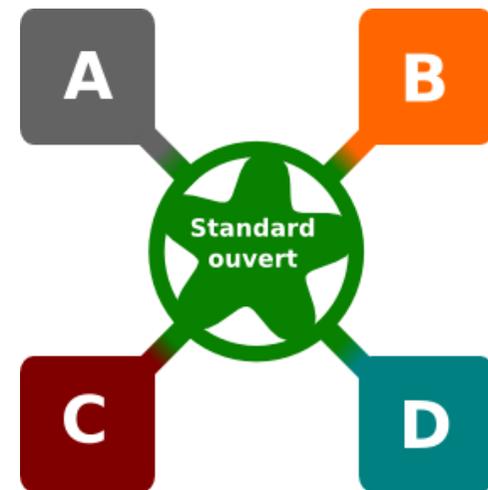
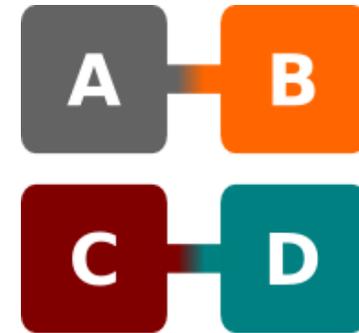
Modèle économique de la création logicielle (2)

- Pendant ce temps, l'innovateur original peut progresser
 - S'il ne le fait pas, il sera dépassé par les innovations des autres
- Nécessité de traiter spécifiquement le problème des marchés captifs créés par les formats de données
 - Dispositions relatives à la décompilation « à fin d'interopérabilité »



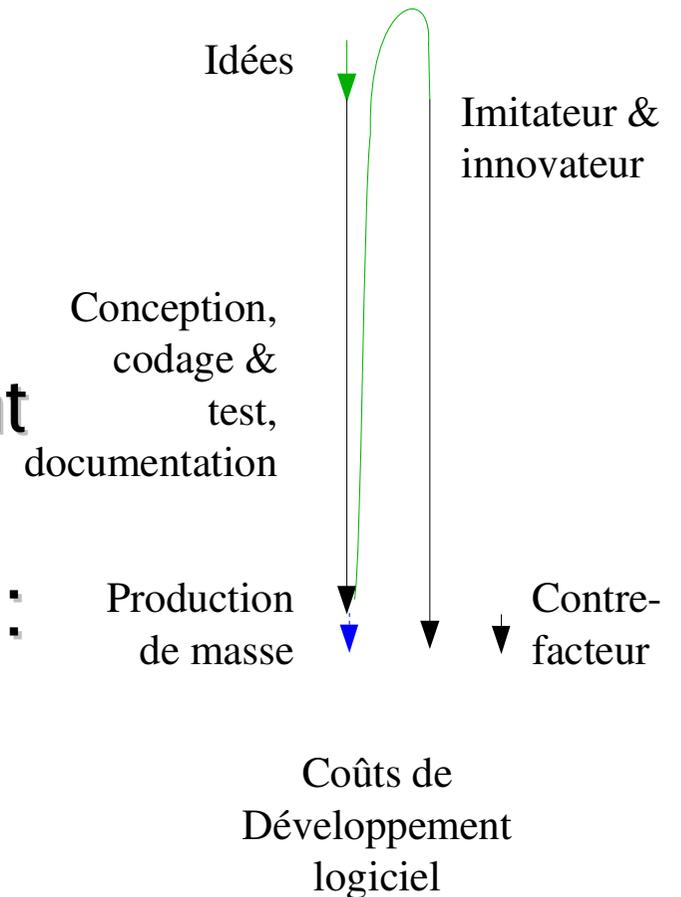
Compatibilité n'est pas interopérabilité !

- Il y a compatibilité quand deux produits peuvent fonctionner ensemble
 - Accord contractuel entre les parties
 - L'entité contrôlant le format « compatible » étend son monopole
- Il y a interopérabilité quand deux produits peuvent fonctionner ensemble et que l'on sait pourquoi



Recherche de l'interopérabilité (1)

- Pour favoriser l'innovation :
 - Les investissements des innovateurs initiaux doivent être récompensés
 - Les nouveaux entrants ne doivent pas être empêchés de concourir
- Pour créer la libre concurrence :
 - La copie servile doit être interdite
 - Le *redesign* doit être légal
- Les marchés captifs doivent être découragés



Recherche de l'interopérabilité (2)

- Spécificité bienvenue du droit européen
 - Instaurée par la directive 91/250/CE
- La décompilation des logiciels est interdite sauf afin de rechercher l'interopérabilité (L.122-6-1 IV° CPI)

Recherche de l'interopérabilité (3)

- Encadrement strict par trois conditions :
 - Actes accomplis par une personne ayant le droit d'utiliser le logiciel ou mandatée à cette fin
 - Informations non déjà disponibles par ailleurs
 - Actes limités aux parties du logiciel nécessaires
- Les informations obtenues ne peuvent être utilisées à d'autres fins et « porter atteinte au droit d'auteur »
- Nécessité de documenter le processus afin de répondre aux accusations de contrefaçon

Statut des interfaces

- La libre concurrence suppose qu'on puisse interfacier un nouveau logiciel avec un logiciel existant
 - Voire le remplacer en s'interfaçant avec les logiciels tiers avec lesquels ce logiciel interagissait
- Or, lorsqu'on veut s'interfacier avec un logiciel existant, il faut respecter ses interfaces
 - Décrites dans des fichiers d'en-tête, des fichiers de classes, etc.
- Pour s'interfacier, il faut donc « recopier » le format des interfaces et leur nommage

Licences

Code is law
Code is poetry
Code is life !

Licence

- La licence est une offre de contrat de la part du fournisseur, qui définit les conditions d'utilisation d'une œuvre
 - Le terme juridique exact est : « pollicitation »
- Basée sur les droits d'auteur ou le copyright
 - Convention de Berne de 1886
- Classiquement, une licence limite les droits d'usage d'une œuvre :
 - Interdiction de diffusion publique
 - Interdiction de reproduction, même partielle

Licences de logiciels (1)

- Il existe de nombreux types de licences de logiciels :
 - « Logiciel privatif » (« propriétaire ») : tous les droits sont réservés par leur titulaire
 - Cas de la majorité des logiciels du commerce
 - Dans la quasi totalité des cas, pour ces logiciels :
 - Le client n'est propriétaire que du support, et pas du logiciel qu'il contient
 - Le fournisseur dégage toute responsabilité en cas de vices cachés (bogues)
 - Non conforme au droit Français
 - Le fournisseur peut arrêter la maintenance du logiciel à tout moment

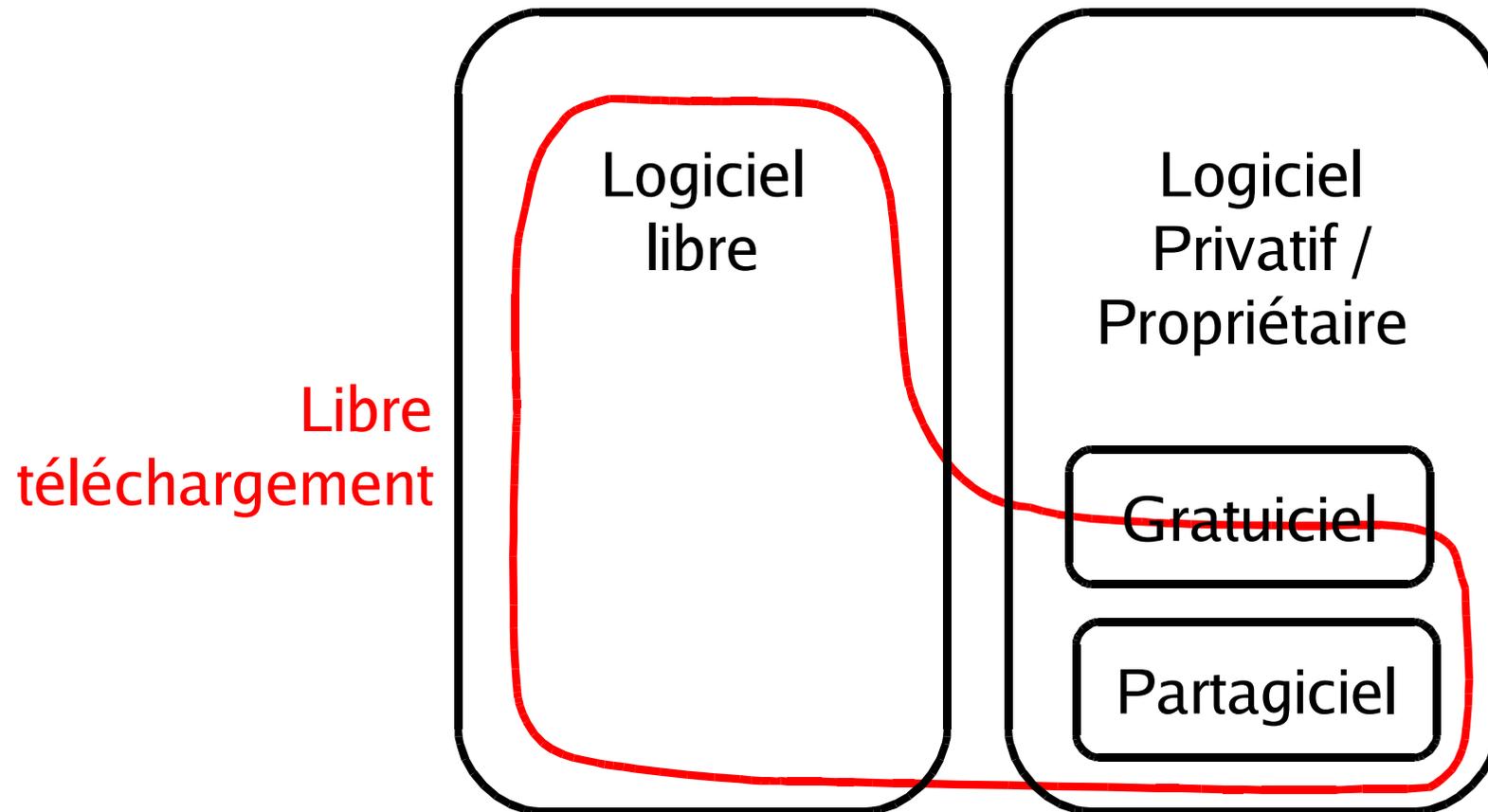
Licences de logiciels (2)

- « Partagiciel » (« *Shareware* ») : logiciel privatif diffusable gratuitement mais pour lequel une contribution est demandée au bout d'une période d'essai
 - Logiciel privatif où seul diffère le mode de distribution
- « Gratuiticiel » (« *Freeware* ») : logiciel privatif gratuit mais ne donnant pas nécessairement d'autres droits
 - Parfois pas même celui de redistribution

Licences de logiciels (3)

- « Logiciel libre » (« *Free software* ») : logiciel donnant de nombreux droits à ses utilisateurs
 - N'est pas équivalent à un « freeware » !
 - Pas nécessairement gratuit
- « de domaine public »

Récapitulatif des types de licences



<http://www.gnu.org/philosophy/categories.html>

Logiciel et licences libres

Free, Free,
Set them free (ouaouuah...)

Logiciel libre

- Le logiciel libre est une innovation juridique et non pas technique
 - S'appuie sur le mécanisme du droit d'auteur applicable aux logiciels
 - Garantit aux usagers des droits et des devoirs
- Permet l'émergence de modèles économiques déconcentrés adaptés à l'économie immatérielle
 - Annule le coût de transaction juridique
 - Coût de transaction technique (coût marginal de copie) rendu nul par Internet

Licences libres (1)

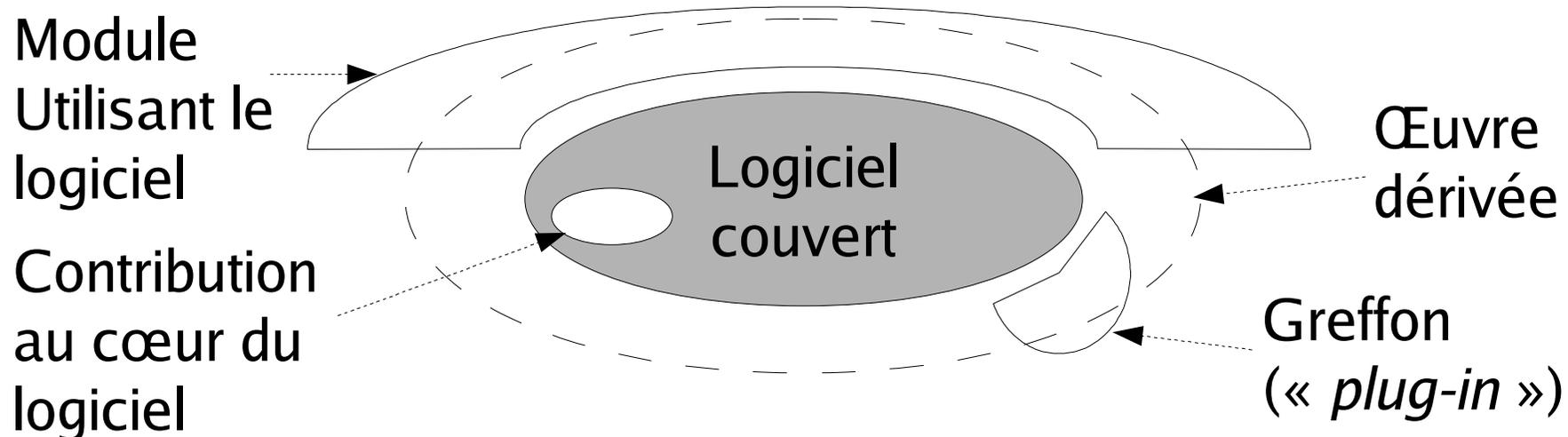
- Ont en commun les « quatre libertés »
 - Liberté d'utilisation pour tout usage
 - Liberté de consultation et de modification du code source
 - Nécessite l'accès au code source
 - Liberté de copie du logiciel original
 - Liberté de rediffusion du logiciel modifié
 - Capitalisation du savoir
 - Mutualisation des développements

Licences libres (2)

- Les différences entre licences libres portent sur les modalités de redistribution du logiciel modifié :
 - Si persistance des obligations de redistribution du code source : « *copyleft* »
 - Liberté collective
 - Si absence de cette obligation : « *non copyleft* »
 - Liberté individuelle

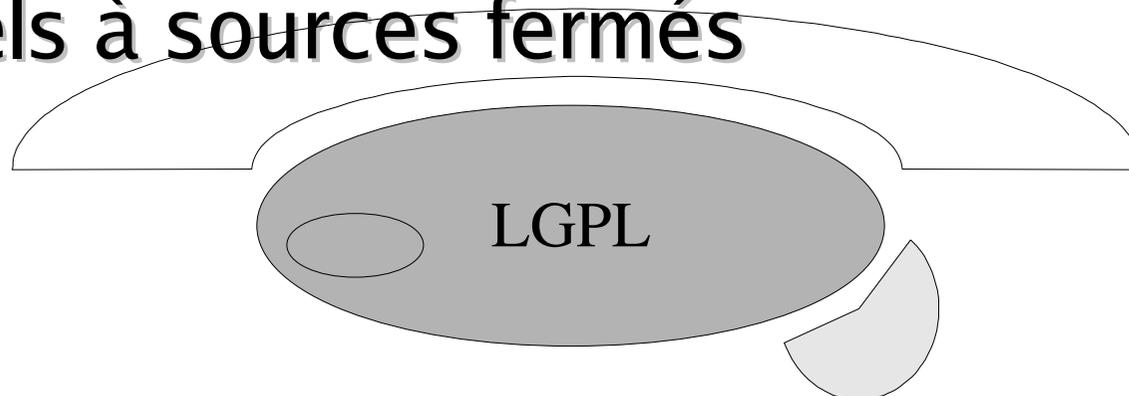
Licences libres (3)

- Trois principaux types de licences libres :
 - Licences « persistantes »
 - Licences « évanescentes »
 - Licences « diffusives »
- Les licences s'appliquent à trois entités, le plus souvent au travers de la notion d'œuvre dérivée



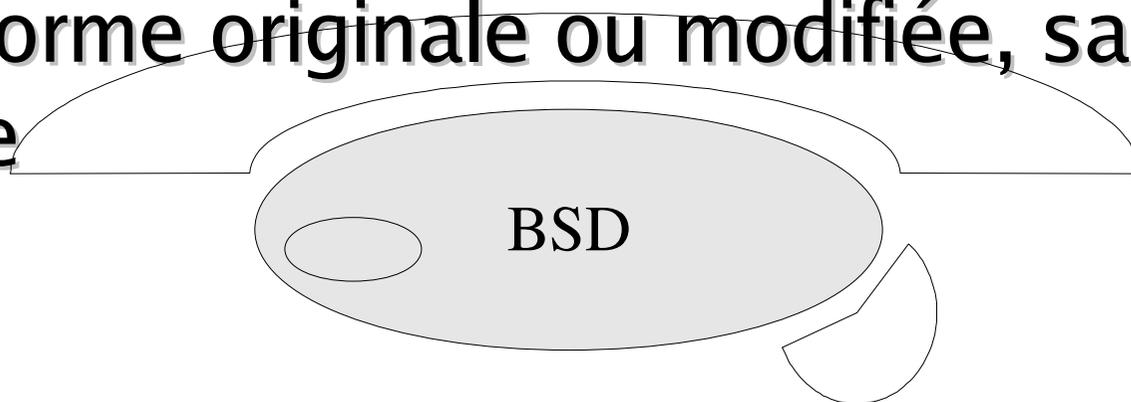
Licences « persistantes »

- Aussi appelées : « pérennes », « à *copyleft* faible »
- Exemples : LGPL, CeCILL-C
- Le code source des versions modifiées doit être rediffusé lorsque celles-ci le sont (pérennité)
- Il peut néanmoins être combiné à des logiciels soumis à d'autres licences, y compris des logiciels à sources fermés



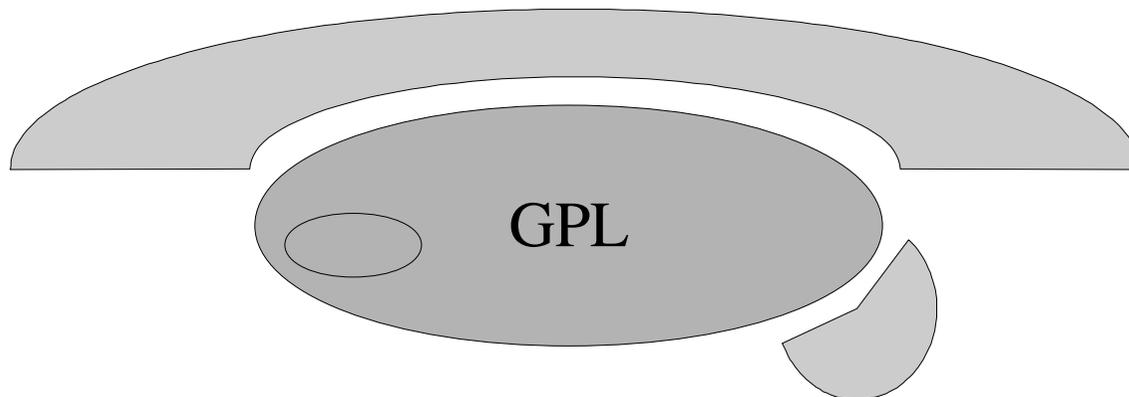
Licences « évanescentes »

- Aussi appelées : « permissives », « non copyleftées »
- Exemples : BSD, CeCILL-B
- Le logiciel diffusé sous forme binaire peut être redistribué selon d'autres termes de licences
 - La CeCILL-B permet de changer la licence du source
- Il peut donc même être « refermé » et distribué, sous forme originale ou modifiée, sans son code source



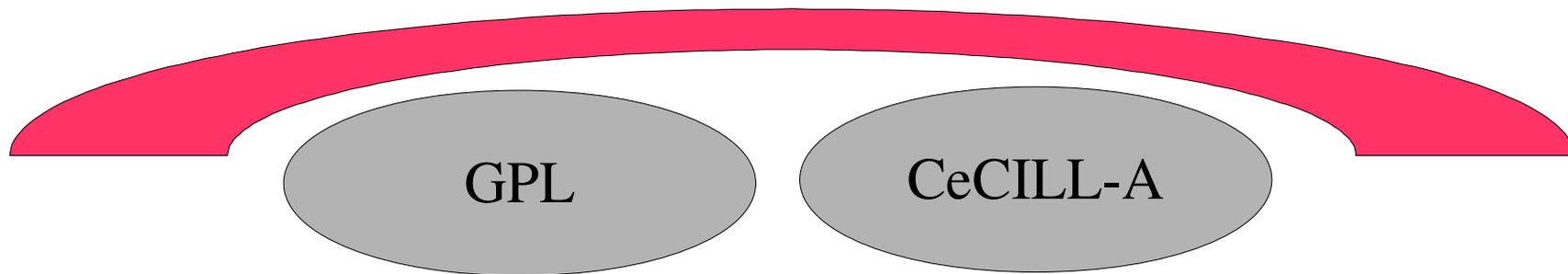
Licences « diffusives » (1)

- Aussi appelées : « à copyleft fort », ou parfois « contaminantes » (terme non neutre, à éviter !)
- Exemples : GPL, CeCILL(-A)
- Le code source des versions modifiées doit être redistribué lorsque celles-ci le sont
- Les logiciels tiers fortement liés doivent être diffusés selon les mêmes termes



Licences « diffusives » (2)

- Les licences diffusives peuvent induire des conflits « en diamant » lorsqu'on cherche à lier ensemble deux modules sous licences diffusives différentes



- Certaines licences « mineures » possèdent des clauses permettant de résoudre ces conflits
 - La CeCILL-A cède explicitement le pas à la GPL

Multi-licenciage

- L'ayant droit d'un logiciel peut choisir de diffuser celui-ci avec le type de licence de son choix
- Il peut même diffuser le même code source, par plusieurs canaux différents, avec des licences différentes
 - Politique de licences multiples : « *dual licensing* »
 - Attention au suivi des versions et contributions successives !

Bonnes pratiques de développement

Mieux vaut prévenir que guérir...

Problématique juridico-technique

- L'analyse juridique doit être concomitante à l'analyse technique
 - Le choix des composants logiciels et de leurs licences conditionne le coût et les modèles économiques du logiciel réalisé
 - Mieux vaut gérer les problèmes en amont qu'en aval
- La compatibilité entre licences doit être étudiée de façon fine
 - Pas nécessairement simple
 - Des logiciels dédiés peuvent (partiellement) automatiser ce processus

Mise en œuvre pratique

- Utiliser des gestionnaires de version pour tracer l'intégralité des contributions
 - Permet de garantir (semi-)automatiquement le respect du droit au nom
 - Permet de savoir quoi re-coder en cas d'incertitude juridique
- Utiliser deux gestionnaires de version :
 - Un gestionnaire privé pour les personnes dont on connaît (et maîtrise) la titularité des droits
 - Un gestionnaire public éventuel pour susciter les contributions de la communauté

Les « brevets logiciels »

On pourrait en rire,
si ce n'était à en pleurer...

Portée du droit d'auteur adapté

- Garantit qu'on ne peut exploiter un programme en dehors des clauses stipulées par sa licence
- N'interdit pas d'écrire un nouveau programme :
 - Aux fonctionnalités similaires
 - Compatible au niveau des formats d'entrée/sortie
 - Interopérable avec le programme original

Protection du logiciel en Europe

- Les programmes informatiques sont protégés par le droit d'auteur
 - Directive 91/250/CE
- Les logiciels sont explicitement exclus du champ de la brevetabilité par l'article 52§2c de la Convention de Munich (1973), tout comme les jeux et les mathématiques

Brevets et logiciels (1)

- Néanmoins, un procédé physique brevetable ne doit pas être exclu de la brevetabilité parce qu'il embarque un logiciel
 - Prise en compte de la numérisation du contrôle des procédés industriels
- Exclusion de brevetabilité limitée au logiciel « en tant que tel »
- Permet de revendiquer des inventions « pilotées par ordinateur »
 - Mais l'innovation ne réside pas dans le logiciel

Brevets et logiciels (2)

- Avec le basculement de l'économie vers le logiciel, les offices de brevets ont vu des pans entiers de l'économie leur échapper
- Tentation de capter à leur profit une partie de cette manne
 - Extension du système de brevets au logiciel
 - Contraire à la volonté du législateur
- Nécessité de tordre la loi, au moyen de deux impostures intellectuelles majeures :
 - Extension abusive du périmètre de brevetabilité
 - Confusion de la fin et des moyens

Critères de brevetabilité

- Pour être brevetable, une innovation doit :
 - Appartenir à un domaine brevetable
 - Domaine « technique », impliquant le contrôle des « forces de la nature »
 - Les pratiques médicales sont exclues de la brevetabilité
 - Être nouvelle
 - Pas de divulgation préalable
 - Démontrer une activité inventive
 - « Non évidente à “l'homme de l'art” »
 - Censé éviter les brevets triviaux
 - Être susceptible « d'application industrielle »
 - Censé limiter le périmètre de la brevetabilité

Extension abusive (1)

- Changement arbitraire des règles d'examen :
 - Anciennes règles : la « nouveauté » ne devait concerner que les revendications « techniques »
 - Nouvelles règles : « nouveauté » considérée « dans son ensemble », donc même si elle ne concerne que les caractéristiques « non techniques »
- Postulat d'existence de logiciels « non en tant que tels » susceptibles d'être brevetés
 - Permet de revendiquer un logiciel nouveau (non « technique ») s'exécutant sur un ordinateur (« technique » mais non nouveau)

Extension abusive (2)

- Terme novlangue d'« invention mise en œuvre par ordinateur »
 - Ce que met en œuvre un ordinateur, c'est un algorithme !
 - Au moyen d'un logiciel qui exprime cet algorithme en tant que création de forme
 - Prétend qu'un algorithme peut être une « invention » au sens de la brevetabilité
 - Alors que les mathématiques ne sont pas brevetables « en tant que telles »

Confusion fins / moyens (1)

- Assertion erronée selon laquelle les logiciels auraient un « effet technique »
 - C'est-à-dire une action dans le monde physique
- Un logiciel manipule des valeurs symboliques
 - Décorrélé de toute réalité physique
 - Exécutable « à la main » ou dans un simulateur sans produire de tels « effets techniques »
- Ce sont les périphériques de l'ordinateur qui interagissent avec le monde physique

Confusion fins / moyens (2)

- L'amélioration d'un logiciel n'est pas une « solution technique » à un « problème technique »
 - Le logiciel ne modifie pas la nature de l'interaction entre l'ordinateur, ses périphériques et leur environnement
- Dans le cas contraire, un pilote de F1 serait brevetable parce qu'il améliore le temps de parcours d'un circuit (« problème technique ») avec une voiture (« moyen technique »)

Confusion fins / moyens (3)

- Changement arbitraire des règles d'examen :
 - Anciennes règles : proposer une « solution technique » à un « problème technique »
 - Nouvelles règles : simple usage de « moyens techniques » pour résoudre n'importe quel type de problème, y compris « non technique »
- Permet de revendiquer les « *business methods* », algorithmes du monde réel :
 - Méthodes de commerce en ligne
 - Calcul de montants de pensions au moyen d'un ordinateur

Portée du brevet algorithmique (1)

- Les brevets algorithmiques ne concernent pas directement les programmes
- Ils visent à monopoliser les concepts sous-jacents, tels que :
 - Ce qu'un programme fait :
 - Quel problème concret ce programme résout, c'est-à-dire quelle « *business method* » il met en œuvre
 - Comment il le fait, et plus particulièrement :
 - Quelles données d'entrée il accepte
 - Quelles données de sortie il produit
 - Comment il interagit avec d'autres programmes

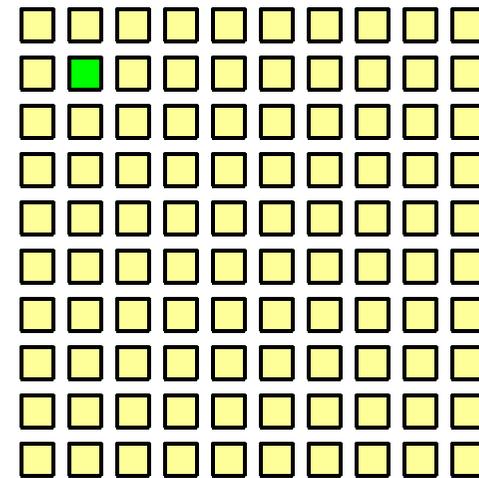
Portée du brevet algorithmique (2)

- Ces brevets portent sur :
 - Les « *business methods* » (« algorithmes du monde réel »)
 - Étapes suivies par un utilisateur achetant des biens sur Internet (brevet « 1-Click » d'Amazon), ...
 - Les formats de fichiers
 - Documents, feuilles de calcul, image (GIF), son (MP3), ...
 - Les protocoles de communication entre programmes
 - Les algorithmes
 - Chiffrement des données, ...
- Ces concepts sont logiquement indissociables

Économie du « brevet logiciel » (1)

- Rêves de fortune

- Mon « invention » brevetée
- Techniques de programmation évidentes

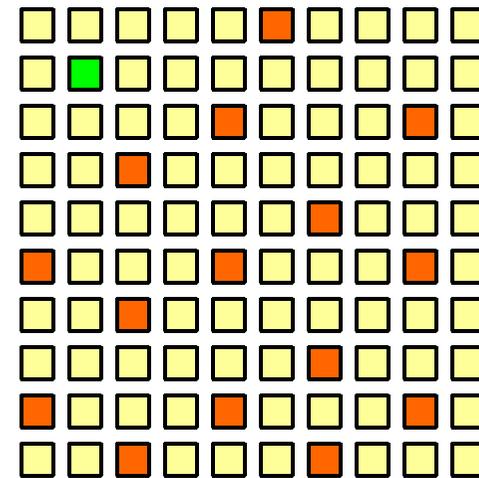


$$\text{Profit} = \text{Ventes} - \text{Develop} + \underbrace{\text{Revenu (■)} - \text{Coût (■)}}_{> 0}$$

Économie du « brevet logiciel » (2)

■ Batailles de brevets logiciels

- Mon « invention » brevetée
- Techniques non brevetées
- Risque de poursuite pour contrefaçon de brevet



$$\text{Profit} = \text{Ventes} - \text{Develop} + \underbrace{\text{Revenu (■)} - \text{Coût (■)} - \text{Coût (■)}}_{< 0}$$

Économie du « brevet logiciel » (3)

$$\text{Profit} = \text{Ventes} - \text{Develop} + \underbrace{\text{Revenu (■)} - \text{Coût (■)} - \text{Coût (■)}}_{\text{En moyenne} < 0}$$

Petit éditeur :

Doit payer

$$\underbrace{\text{Revenu (■)} - \text{Coût (■)}}_{< 0} - \underbrace{\text{Coût (■)}}_{< 0}$$

Gros éditeur :

Neutralise les brevets concurrents

$$\underbrace{\text{Revenu (■)} - \text{Coût (■)}}_{< 0} - \underbrace{\text{Coût (■)}}_{\approx 0}$$

Fonds de brevets :

Vit sur le dos du système

$$\underbrace{\text{Revenu (■)} - \text{Coût (■)}}_{\text{En principe} > 0} - \underbrace{\text{Coût (■)}}_{= 0}$$

Qui survit ?

- Les grands groupes
 - Échanges de portefeuilles de brevets
 - Position de quasi-monopole
- Les fonds de brevets
- Les petites entreprises désirant se faire racheter
 - Leurs brevets augmentent leur valeur d'achat supposée...
 - Mais risque systémique ignoré !
- Les avocats, les experts en propriété industrielle, les offices de brevets
 - Consomment entre 10 et 40 % des moyens

Qui déperit ?

- Les petits éditeurs de logiciels, généralement peu préparés aux batailles juridiques, par :
 - Manque de fonds (coût moyen d'un procès > M\$)
 - Manque de portefeuilles de brevets à échanger
- Les entreprises utilisant du logiciel « maison »
 - Même si ces entreprises n'appartiennent pas au monde du logiciel
 - Même si elles ne les commercialisent pas
- Les développeurs de logiciels libres
 - Code source mis à disposition de la partie adverse

Menaces sur l'innovation et l'interopérabilité

- Si le détenteur d'un brevet algorithmique refuse de céder des licences :
 - Il est illégal d'écrire des programmes qui lisent ou produisent des formats de données brevetés
 - Les utilisateurs ne peuvent plus basculer vers d'autres produits pour traiter leurs données existantes
 - L'entrée sur le marché de nouveaux acteurs est réduite
 - Le choix des produits et des fournisseurs est réduit

Bilan du brevet algorithmique

- Instrument conçu pour l'industrie matérielle et étendu abusivement aux méthodes intellectuelles
 - Monopolise les algorithmes (= mathématiques !)
- Illégal en Europe mais accordé par les offices, qui vivent des annuités
- Contresens économique
 - Instrument anti-concurrentiel
 - Coûteux à obtenir et à maintenir
 - Menace le retour sur investissement
 - Favorise les délocalisations

Conclusion

Conclusion

- Les informaticiens ne doivent pas méconnaître le droit
 - Inclusion de paramètres juridiques dans tous les critères de décision, à toutes les étapes du cycle de vie d'un projet
- Les juristes ne devraient pas non plus méconnaître l'informatique...
 - Aberrations telles que le brevet algorithmique
- Les informaticiens doivent pouvoir éclairer le législateur