

# Brevets et droit d'auteur

*Ce qu'un informaticien ne peut pas ne pas savoir*

Thomas Tempé

`Thomas.Tempe@insa-lyon.fr`

INSA de Lyon, dpartement Informatique

# La propriété intellectuelle en France

- Les marques (on n'en parlera pas ici)
- Le droit d'auteur
- Les brevets

# Le droit d'auteur

# Le droit d'auteur

*Copyright* en anglais.

- Protège une œuvre
- Monopole de l'auteur sur l'exploitation de l'œuvre
- Très longue durée (70 ans après la mort de l'auteur)
- Attribué automatiquement, dès qu'il y a création

# Le droit d'auteur protège

- Les livres
- Les partitions musicales
- Les tableaux
- Les programmes informatiques
- ...

# Licence et droit d'auteur

L'auteur décide des droits qu'il vous donne :

- la permission d'écouter ?
- la permission de redistribuer ?
- la permission de modifier ou désassembler ?

... et vous fait éventuellement payer pour ces droits.

# Freeware, shareware et domaine public

Quels droits avez-vous ?

- **Domaine public** : les œuvres qui ne sont *plus* soumises au droit d'auteur (classiques de la littérature...)
- **Shareware** : l'auteur vous vend le droit d'utiliser le programme
- **Freeware** : pas de définition unanime

# Logiciel libre

- Il *reste propriétaire* de l'œuvre
- Il *met une licence* sur son œuvre, qui garantit les 4 libertés fondamentales :
  - Le droit d'utiliser, pour tous
  - Le droit de redistribuer (ou revendre)
  - Le droit d'étudier et de modifier
  - Le droit de redistribuer une version modifiée

# Logiciel libre

Attention ! Si vous utilisez du code libre dans votre travail, **il reste la propriété de son auteur.**

- il faut citer son nom
- il faut respecter la licence originale
- l'auteur peut interdire de mélanger son programme à du code non-libre.

# Logiciel sans mention de l'auteur

En France, le droit d'auteur s'applique automatiquement.

- On ne sait pas quels droits on a :
  - **on n'en a aucun !**
- On ne connaît pas l'adresse de l'auteur :
  - **On ne peut pas en demander !**
- On ne connaît pas la date de création :
  - **On ne sait même pas quand le droit d'auteur expire...**

# Conclusion sur le droit d'auteur

- On ne peut pas faire n'importe quoi avec une œuvre.
- Dans une entreprise, le non respect de ces droits peut poser de gros problèmes.
- Il est essentiel d'avoir un minimum de connaissances.
- On ne peut pas faire n'importe quoi avec du logiciel libre.

# Les brevets

# Qu'est-ce qu'un brevet

- Monopôle sur l'exploitation d'une idée
- Nécessite un dépôt de dossier (long et coûteux)

<b>Droit d'auteur</b>	<b>Brevet</b>
protège une œuvre	protège une invention
dure > 70 ans	dure 10 ou 20 ans
ex : un livre	ex : une méthode d'impression

# Conditions préalables

Pour qu'un brevet soit valable, une invention :

- Doit être novatrice
- Doit être exploité
- La recette doit être divulguée

# Utilité du brevet

- Une entreprise investit de grosses sommes pour développer un produit. Un concurrent ne pourra pas le copier impunément dès sa sortie.
- Une entreprise a développé un nouveau composé chimique, elle n'en gardera pas la formule secrète indéfiniment.

# Exemple de brevets

- une formule particulière pour un acier spécial
- une brique combinant structure (béton) et isolant (polystyrène)
- le téléphone
- une molécule
- un tire-bouchon ingénieux

# A quoi servent les brevets (dans la vraie vie)

- **À s'assurer un monopole :**
  - Microsoft avec Palladium, les fabricants de médicaments
- **À monnayer/contrôler l'emploi d'une technologie :**
  - Le consortium MPEG
- **À effectuer des échanges stratégiques :**
  - IBM déclare que c'est leur principale utilité

# Comment déposer un brevet ?

- Rédiger une déclaration avec l'aide d'un avocat
- Effectuer une demande auprès de l'Office Européen des Brevets
- La demande dure env. 2 ans

# Brevets sur les logiciels

# Le problème des brevets sur les logiciels

- Jusqu'à présent, on ne pouvait pas breveter une idée, seulement une invention technique (réalisable)
- Aux États-Unis et au Japon, les idées et les algorithmes sont aujourd'hui brevetables.
- En Europe, des pressions politiques tentent d'imposer les brevets sur les logiciels, face à une vive opposition.
- Exemple de brevet sur un logiciel : brevet sur une architecture de système d'exploitation temps réel.

# Que protège un brevet sur un logiciel ?

Il protège une idée, générique.

- Que diriez-vous si quelqu'un avait breveté :
  - le diagnostic médical assisté par ordinateur ?
  - l'application du *ou exclusif* sur une image noir-et-blanc ?
  - l'enseignement assisté par ordinateur de la prononciation de langues étrangères ?
  - le panier électronique (*shopping cart*) ?
  - le contrôle d'un ordinateur par un autre ?

**Toutes ces idées ont été brevetées**

# En quoi le logiciel est-il différent ?

- L'idée ne coûte rien, c'est la mise en œuvre qui est chère
- il faut **beaucoup** d'idées pour faire un programme
- en 20 ans, une idée nouvelle est devenue éculée
- impossible de faire des recherches d'antécédents
- Le logiciel est ce que l'homme a créé de plus complexe
  - pas besoin de re-fabriquer la roue
  - pas de problèmes de vibration, d'usure

# Les problèmes génériques des brevets

- Les brevets ont un spectre de plus en plus large, couvrant des choses pas imaginées par leurs auteurs
  - brevet sur l'hypertexte par British Telecom, en 1989.
  - les *moulins à brevets*
- Les examinateurs acceptent des brevets qui n'ont rien de novateur
  - Ex : brevet sur les 35 heures par l'AFUL, sur la roue par un avocat australien
- Les procès pour violation de brevets illégitimes sont très coûteux, et la plupart des conflits sont résolus "à l'amiable"

# Autres problèmes liés aux brevets logiciels

- Les brevets peuvent nuire à l'interopérabilité des systèmes
  - brevets sur des standards
- Les brevets sont incompatibles avec le logiciel libre
- Les brevets US ne sont pas publics pendant leur période d'évaluation
  - les idées brevetées peuvent être devenues triviales entre-temps
- Des brevets souterrains peuvent ressurgir des années après l'adoption généralisée d'une technologie
  - Ex : le mp3, le Gif...

# Insécurité juridique

Il est impossible de se prémunir contre les brevets logiciels, parce que :

- Même ce qui est évident peut être breveté
- Les brevets sont rédigés dans un langage délibérément incompréhensible
- On peut discuter longtemps avant de savoir ce qui est couvert ou non par un brevet
- Certains brevets couvrent tout et n'importe quoi
- Le dépôt de brevet peut être postérieur à l'invention

... et il est très coûteux de se défendre suite à un procès, même illégitime.

# À qui profitent les brevets logiciels ?

- Pour les grandes entreprises : ils permettent d'effectuer des accords croisés stratégiques
- Pour les PME/PMI, ils sont un risque impossible à gérer, et n'ont que peu d'avantages
- Pour le Logiciel Libre : il est souvent impossible d'implémenter une idée brevetée.

# La situation des brevets logiciels en Europe

- L'OEB (*Office Européen des Brevets*) accepte illégalement les dépôts de brevets sur des idées et algorithmes depuis quelques années.
- La BSA (*Business Software Alliance*) et l'OEB font pression sur le parlement européen pour légaliser ces pratiques.
- Beaucoup d'entreprises (surtout américaines) déposent de tels brevets en Europe, en attendant cette légalisation pour qu'ils deviennent applicables.

# Illustration du danger des brevets

- une start-up invente une méthode de compression vidéo fantastique, pour regarder des films par modem.
- à grand frais, elle dépose un brevet, puis développe un produit.
- Une grosse entreprise la remarque, propose de la racheter et essuie un refus.
- L'entreprise l'attaque alors pour violation de 12 brevets.
- Incapable de se défendre, la start-up accepte un règlement à l'amiable : 5% du CA pour chaque brevet.
- La start-up va couler, mais heureusement se fait racheter par l'entreprise, qui s'accapare ainsi les brevets.

# Conclusion

- Les logiciels sont protégés par le droit d'auteur
- Les inventions sont protégées par les brevets
- La menace des brevets sur les logiciels plane sur l'Europe, et en particulier
  - sur les PME
  - sur le logiciel libre

# Conclusion

Allez vous renseigner, et signer la pétition :  
<http://www.eurolinux.org>

# Redistribution de ce document

Copyright 2003 © Thomas Tempé.

Permission est accordée de distribuer et modifier ce document selon les termes de la licence

*GNU Free Documentation Licence* dans sa version 1.1 ou postérieure, telle qu'elle est publiée

par la *Free Software Foundation*