

**Le logiciel libre est-il soluble dans  
l'économie marchande?  
Typologie des modes d'hybridation des  
logiciels libres et propriétaires**



***Claire Scopsi***  
*Maître de conférences, INTD-CNAM*

[claire.scopsi@cnam.fr](mailto:claire.scopsi@cnam.fr)

# Sommaire

## **1 Quelques définitions**

1.1 Naissance d'un concept

1.2 Libre, Open source, propriétaire, gratuit...

1.3 Ce que je peux faire techniquement et légalement avec un Logiciel Libre

## **2 Situation actuelle des logiciels libres**

2.1 Quelques chiffres

2.2 Une phase de maturité.

2.3 Nouvelles communautés, nouveaux acteurs

## **3 Modèles d'hybridation**

3.1 Délégation de compétences

3.2 Licences hybrides

3.3 Logiciels hybrides

**Concurrence ou complémentarité des logiciels libres et propriétaires?**

# 1 Quelques définitions

# Naissance d'un concept

## Naissance d'un concept

1969 Création du système d'exploitation UNIX

1973 Diffusion d'Unix dans les universités américaines, qui corrigent le programme.

1982 Commercialisation d'UNIX par AT&T

1983 Richard Stallman lance le projet GNU (Unix libre)

1985 Richard Stallman conceptualise le « libre » en créant la Free Software Foundation

1991 Linus Torvald lance le projet Linux

1992 – 2003 Série de procès entre les sociétés commerciales détentrices de code UNIX

1997 Création de l'Open Source Initiative par Bruce Perens, Eric Steven Raymond

# Logiciels libres et open source

Le **Logiciel libre** répond aux 4 critères de la Free Software Foundation :

- la liberté d'exécuter le programme pour tous types d'usages.
- la liberté d'accéder au code source, de l'étudier, de l'adapter.
- la liberté de distribuer des copies du programme, modifié ou non, gratuitement ou non
- la liberté d'améliorer le programme et de publier ces améliorations.

Le **logiciel open source** précise et cadre les critères des logiciels libres, dans le contexte d'une compatibilité avec une exploitation commerciale:

1. Libre Redistribution
2. Code Source
3. Applications Dérivées
4. Intégrité du Code Source de l'Auteur
5. Pas de Discrimination Contre des Personnes ou des Groupes
6. Pas de Discrimination Contre des Domaines d'Application
7. Distribution de la Licence
8. La Licence ne doit pas être Spécifique à un Produit
9. La Licence ne doit pas Affecter d'Autres Logiciels
10. La Licence doit être Technologiquement Neutre

***Les licences libres précisent les conditions d'utilisation des LL dans le cadre défini par l'OSI. L' OSI évalue et certifie la conformité de ces licences avec son modèle.***

# Logiciels propriétaires, logiciels compilés

## Logiciel propriétaire :

Un logiciel qui ne répond pas à toutes les conditions du logiciel libre est dit « propriétaire ».

## Logiciel compilé :

De nombreux programmes sont livrés sous une forme compilée, i.e. une traduction des sources dans un autre langage afin:

- de permettre une meilleure exécution
- d'interdire la lecture des sources.

Pour mériter l'appellation « libre » un tel logiciel doit fournir ses sources sous une forme accessible en plus de la version compilée.

# Gratuits

Un **Freeware** (gratuiciel) est toujours gratuit, mais son code n'est pas toujours ouvert.

**Ne pas confondre Free Software (logiciel libre) et freeware (logiciel gratuit). Les logiciels libres ne sont pas obligatoirement gratuits.**

**(même si de fait ils le sont souvent!).**

# Ce que je peux faire avec un logiciel libre

## Ce que je peux faire techniquement :

- étudier le code source pour en comprendre la logique
- copier des parties de code pour faire un autre logiciel
- corriger des bugs
- ajouter des fonctions manquantes
- améliorer les fonctions existantes
- l'associer à un autre code
- supprimer une partie du code...

## Ce que je peux faire légalement :

- Utiliser le programme dans mon activité professionnelle, encours, le donner à des élèves, à des clients
  - le vendre (sans reverser de droits d'auteurs) associé à d'autres logiciels ou intégré dans le code d'un autre produit , ou encore en version modifiée
- dans le cadre défini par la licence .**

***Si on associe des codes sources d'origines diverses, il faut donc faire attention à ce que l'utilisation soit conforme à toutes les licences.***

## **2 Situation actuelle**

# Quelques chiffres

**170 000 projets** référencés sur le site Sourceforge.net.

Le patrimoine d'applications dites "libres" ou "open source", créées par une communauté active, représente l'équivalent de **131 000 années/hommes** ? (*lu dans le rapport Attali*)

Part de marché français encore faible (**2,1%** estimés en 2007) mais en forte croissance (**+63%**)

Club Kohala (utilisateurs de Koha en France) : une **trentaine de membres**.

**300 inscrits** à la liste des utilisateurs PMB.

Selon l'enquête de Tosca Consultants sur les SIGB, Koha aurait eu **28 clients**, et PMB **600 nouveaux clients**, en 2007 sur un marché français d'environ 18 000 clients. (enquête reposant sur la déclaration de l'éditeur ou du leader de la communauté). **Un projet de SIGB sur 5 serait un projet open source.**

**26 779 membres** sur le forum français de Joomla, dont 1741 actifs

# Une phase de maturité

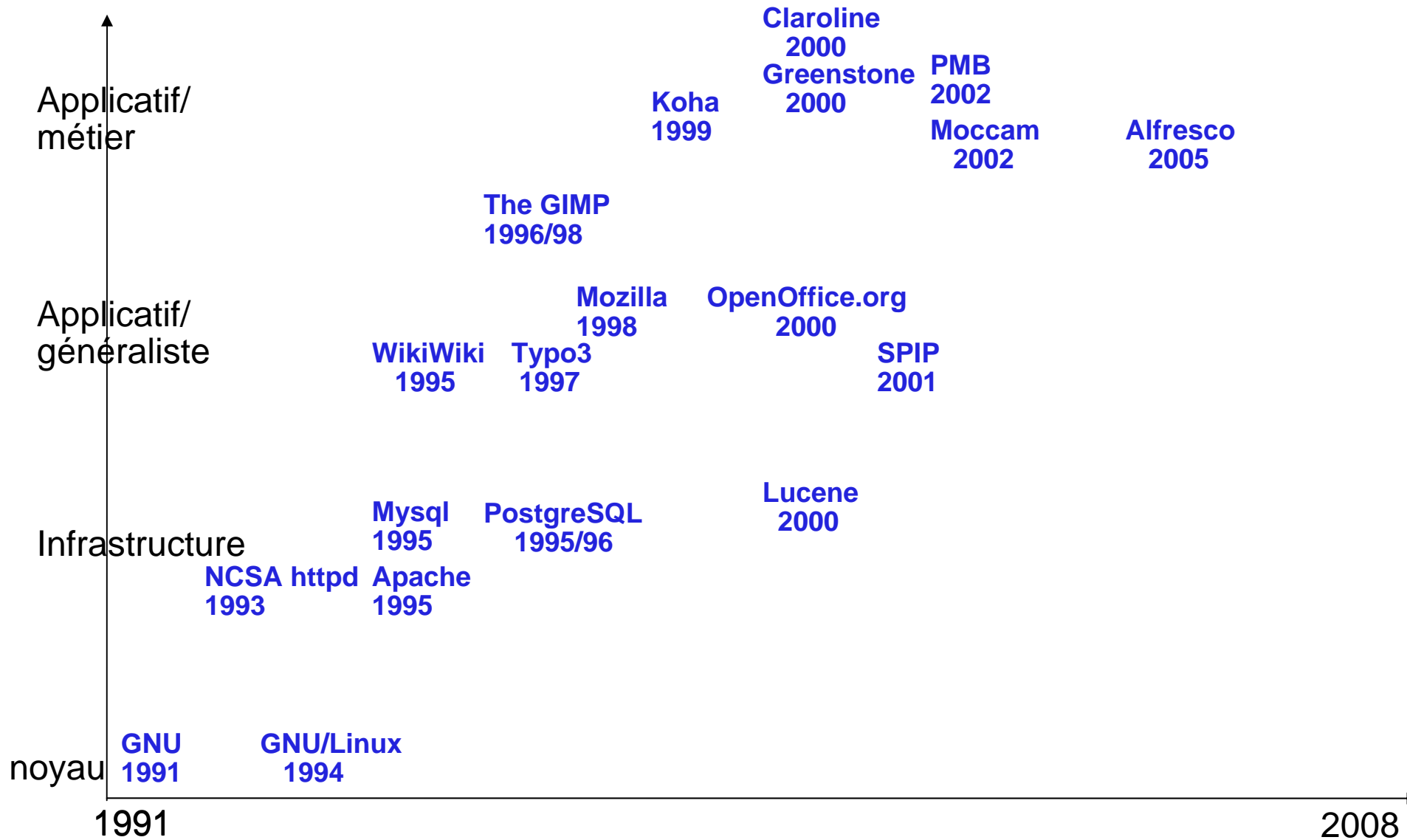
mise en place d'un écosystème d'entreprises permettant de répondre à toutes les attentes

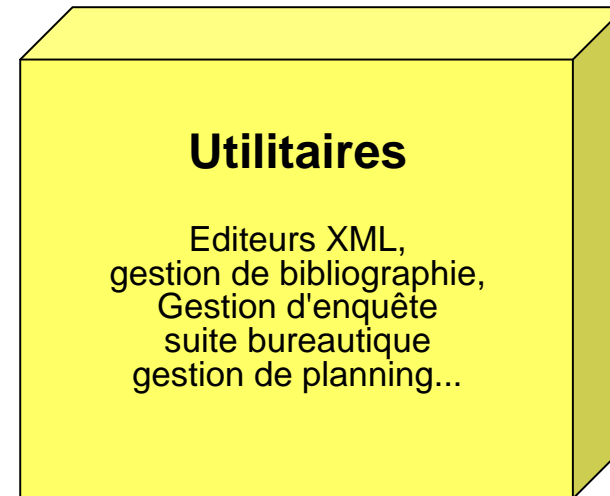
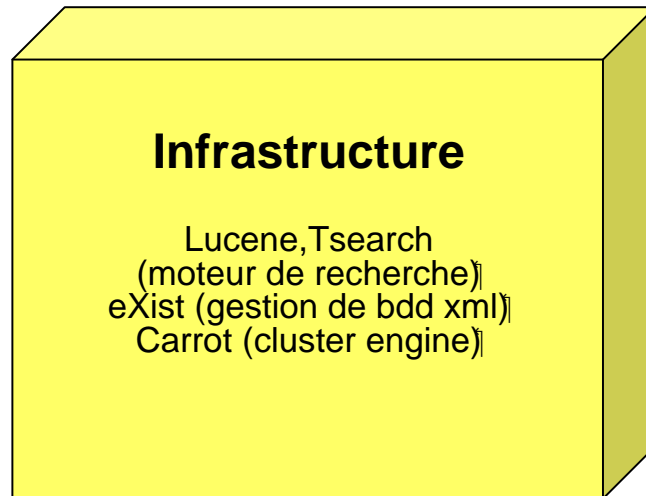
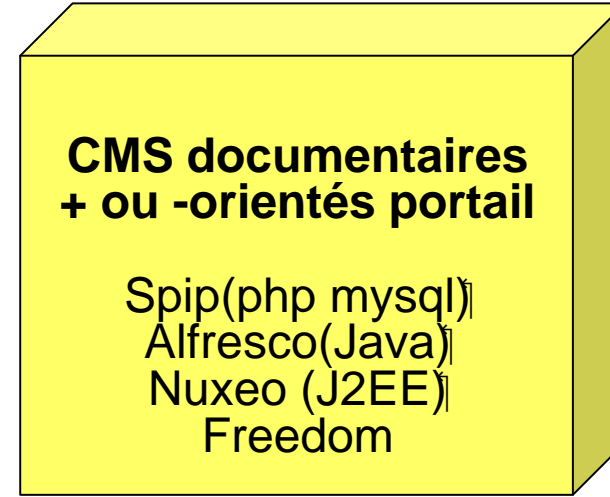
atténuation des discours militants, émergence d'une communication sur les « business models »

meilleure prise en compte des attentes en matière d'interfaces ou de « finitions »

la confiance des entreprises est gagnée

la couverture logicielle est à peu près complète





# Nouvelles communautés, nouveaux acteurs

## **Caractéristiques des communautés d'infrastructures :**

- nombre de membres important
- membres techniciens
- contributeurs / utilisateurs
- fonctionnalités homogènes
- utilisateurs autonomes
- contextes d'utilisation divers (y compris recherche, bénévolat)

## **Caractéristiques des communautés des applications métier :**

- nombre de membres réduit
- membres non techniciens
- développeurs non praticiens
- fonctionnalités : diverses, mouvantes
- utilisateurs en attente d'assistance
- contexte d'utilisation exclusivement professionnel

## **Paradoxe du libre (pour l'utilisateur métier) :**

Comment exercer la "liberté d'accéder au code source, de l'étudier, de l'adapter » quand on n'est pas informaticien?

Comment rassurer l'entreprise et lui donnant des garanties d'assistance, de correction, d'évolution?

## **Paradoxe du libre (pour la communauté de développeurs)**

:

Comment « faire le job » quand il y a plus de prescripteurs que de développeurs dans la communauté et qu'il faut en outre assurer une assistance à l'utilisation?

## **3 Modèles d'hybridation**

# Délégation de compétences

## **Paradoxe du libre (pour l'utilisateur) :**

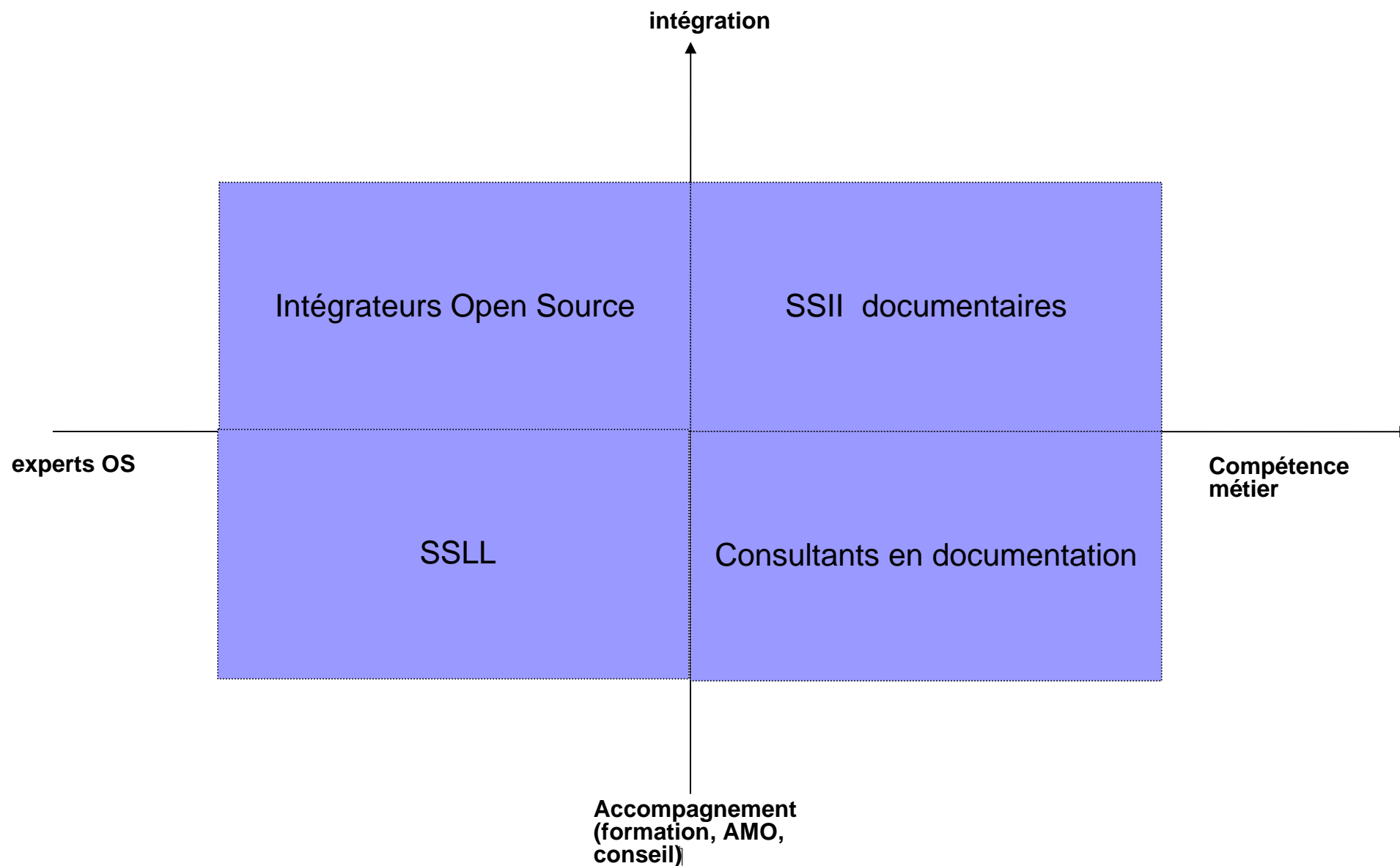
Comment exercer la "liberté d'accéder au code source, de l'étudier, de l'adapter » quand on n'est pas informaticien?

En louant les compétences d'un tiers technique.

Comment rassurer l'entreprise et lui donnant des garanties d'assistance, de correction, d'évolution?

En contractant avec un tiers technique, ou avec une structure issue de la communauté.

# Délégation de compétences



# Les licences

## **Paradoxe du libre (pour la communauté de développeurs)**

:

Comment « faire le job » quand il y a plus de prescripteurs que de développeurs dans la communauté et qu'il faut en outre assurer une assistance à l'utilisation?

*En troquant le modèle du bénévolat contre un modèle marchand pour rémunérer les techniciens.*

# Les licences

**Licences duales** : versions gratuites ou payantes selon les usages (mysql)

**Licences chronodégradables** : la licence « sort sous une forme libre un an après la sortie de la licence propriétaire.

**Libération stratégique** ou « dynamitage » : la licence est libérée pour accélérer la diffusion (s'imposer comme standard)

**La licence libre « carte de visite »** : l'image originale ou positive du libre, rejaillit sur les prestations de services ou d'autres produits associés.

**Logiciels libres « financés par des communautés de sociétés commerciales »** : le groupement favorise la multiplication de logiciels partageant leur stratégie, pour faire basculer le marché.

# Les logiciels

**Livraisons mixtes** : dans une même livraison applicative sont associés des logiciels libres et des logiciels sous licences propriétaires.

**Logiciels hybrides** sur le critère de la **valeur ajoutée** de l'éditeur : les couches d'infrastructures (sgbd, moteur de recherche...) et intégrées aux couches métier ou innovantes relevant d'un savoir faire spécifique de l'éditeur. L'ensemble est sous licence propriétaire.

**Logiciels hybrides** sur le critère de la **valeur ajoutée** : partenariat entre une communauté et un éditeur associant deux compétences métier.

## Concurrence ou complémentarité des logiciels libres et propriétaires?

Concurrence? oui

Complémentarité? oui

*Mais ces deux situations « dichotomiques » s'effaceront progressivement au profit d' une gamme d'offres continue du libre « pur » au « propriétaire « pur » en passant par toute une combinaison de solution mixtes.*

Le paysage futur offrira plus de choix, mais sera plus complexe.