



Présentation



JANUA

OPEN SOURCE & OPEN STANDARDS

Pascal FLAMAND
pflamand@janua.fr
0 950 260 370



Comprendre l'Open Source et ses enjeux, les risques et les opportunités du choix de l'Open Source en entreprise.

1- Présentation de Janua

2 – Modèles économiques

3 - Panorama de l'offre logicielle O.S. Actuelle

4 - Risques et opportunités du choix de l'Open Source en entreprise

5 - Réussir une implémentation Open Source

Présentation De Janua

- Société de services (SS2L)/éditeur fondée en 2004 à Sophia Antipolis
- Notre métier : l'expertise
- Notre crédo : l'Open Source
- Notre force : les hommes, leur vécu et leur motivation
- Notre approche : les solutions packagées et les POCs
- Nos prestations : consulting, assistance ponctuelle et développement au forfait

Les compétences de Janua

- Consulting, implémentations et déploiement de solutions de gestion des identités et de messagerie d'entreprise.
- Etudes d'opportunités et accompagnement à la migration Open Source, déploiement de solutions d'infrastructures.
- Développement au forfait, assistance et expertise sur des composants Open Source et embarqués.
- Consulting réseaux et sécurité, ISO 27001, expertise base de données, cluster et virtualisation..
- Editeur des logiciels CmakeBuilder, Jaguards, JAC, KressourcesWCAP et EZ-Slony.

Gestion de l'identité et messagerie

- Du consulting au déploiement :
 - SSO et PKI
 - Annuaires LDAP
 - Gestion d'annuaires
 - Workflow et provisioning d'identités
 - Migrations NIS vers LDAP, Synchro AD<=>LDAP
 - Messagerie d'entreprise



Migration Open Source

- Etude d'opportunités et accompagnement à la migration Open Source (SGBD, CMS, Open Office..)
- Virtualisation, clusters, sécurisation d'infrastructures, procédures d'exploitations
- Expertise noyau et outils Linux/Open Solaris, développement de distributions embarquées et drivers spécifiques
- Management et infrastructure Réseau Open Source, développement d'agents SNMP et de MIB spécifiques
- Les solutions Open Source "bundlées" par Janua: Janua Development Center et Janua Helpdesk Center

Développement au forfait

- Bases de données : MYSQL, POSTGRES, SLONY-I
- Outils : C, C++, PERL, JAVA, J2EE, PHP, Zope, Jabber/XMPP, Glassfish
- Réseaux : NetSNMP, Nagios, agents SNMP, MIBs
- Systèmes d'exploitations : UNIX, LINUX, distributions embarquées et drivers spécifiques, J2ME
- Janua est l'éditeur des logiciels CmakeBuilder, Jaguards, JAC, EZ-Slony et KRessourcesWCAP

Consulting

- Identity Management, annuaires, SSO et PKI
- Audit, migration et déploiement de messageries d'entreprise
- Virtualisation, PRA, procédures d'exploitation
- Conseil et déploiement de CMS
- Expertise noyau et outils Linux/Open Solaris
- Audit d'infrastructures réseaux
- Conseil, audit et mise en oeuvre d'infrastructures de développement optimisées: CVS et SVN, simulation et cross compil, bug-tracking ,inventaire et gestion d'incidents, Eclipse et Netbeans.
- Certification et audit ISO 27001



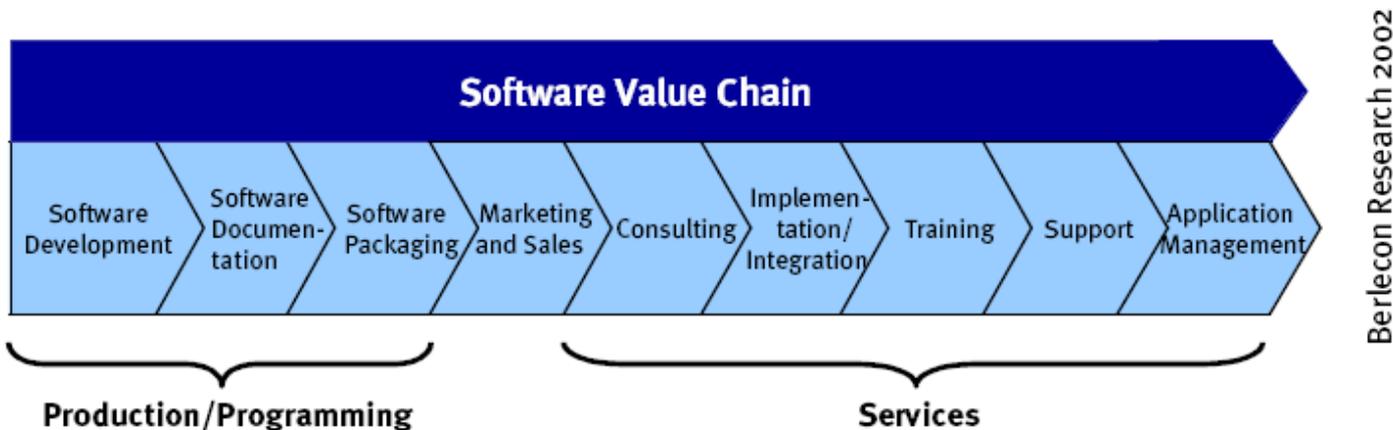
Les modèles économiques

Création et captation de valeur dans l'industrie logicielle

- Singularité des biens numériques :
 - Coûts variables de reproduction proche de zéro,
 - Coûts fixes de développement très élevés.
- Sur l'ensemble du cycle de vie d'un logiciel, il existe
 - Des points de création de valeur
 - Des points de captation de valeur

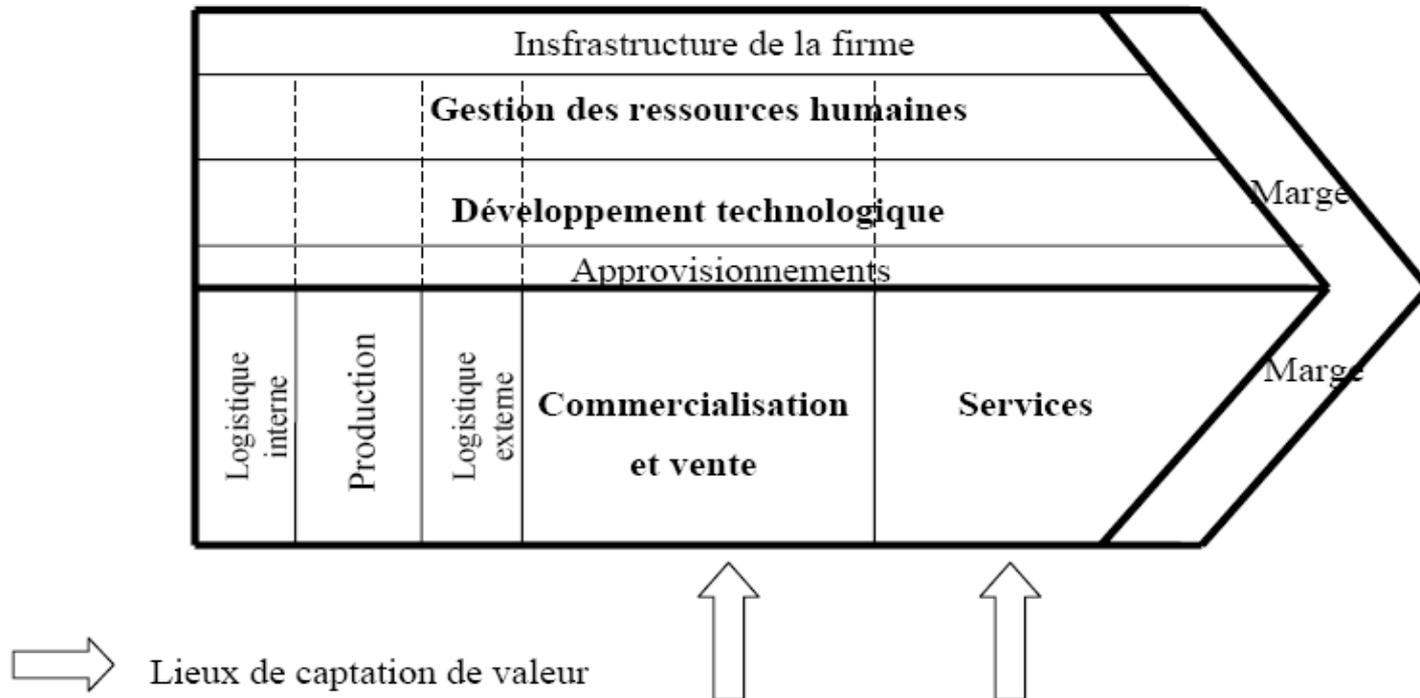
Dans le modèle traditionnel (propriétaire)

- Création de valeur :
 - Essentiellement en interne (chez l'éditeur)



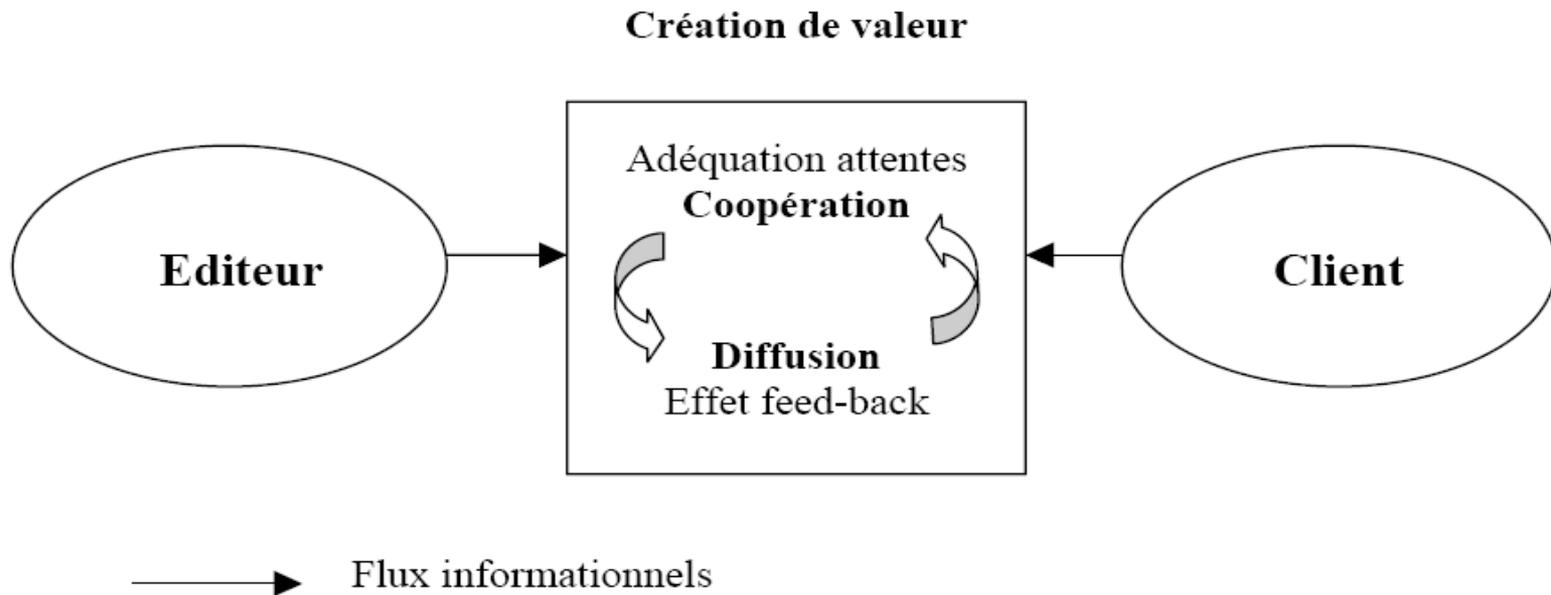
Dans le modèle traditionnel (propriétaire)

- Captation de valeur :



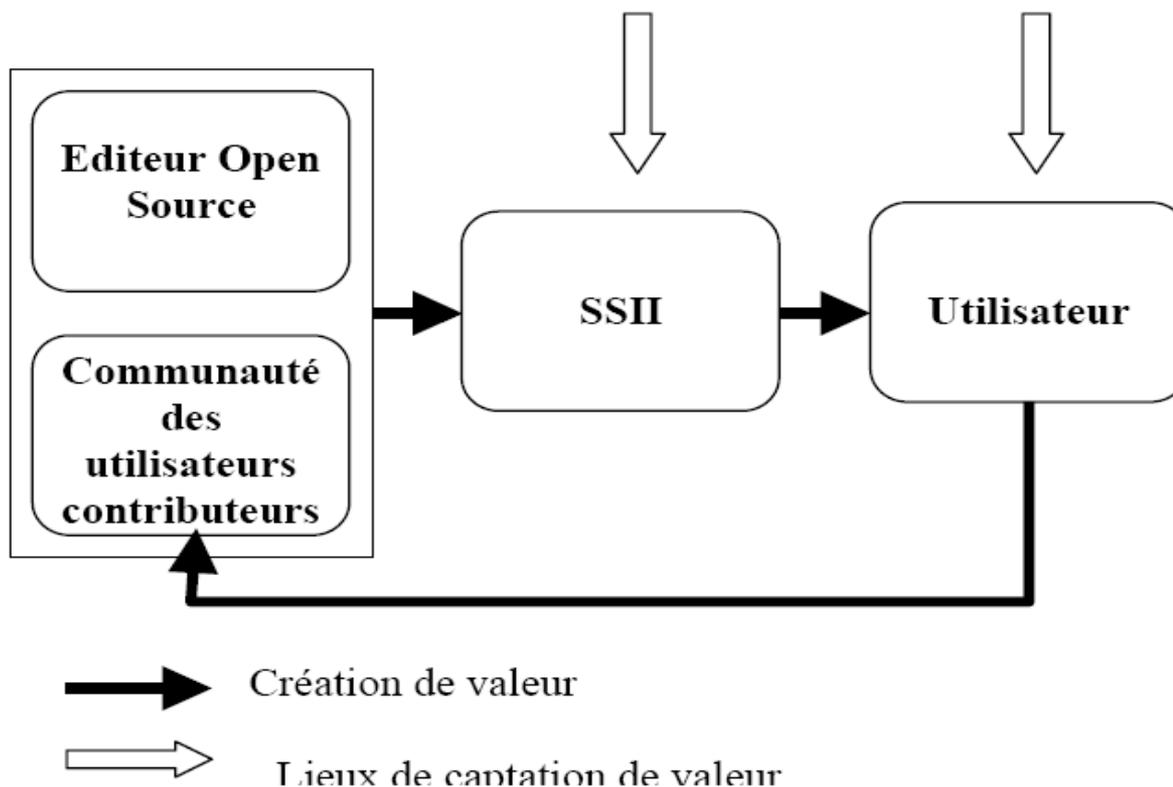
Dans le modèle Libre

- Création de valeur :
 - Fait appel à des ressources externes (coopération)



Dans le modèle Libre

- Captation de valeur :



Synthèse sur l'analyse de la chaîne de valeur

- Le modèle libre est encore jeune
- Encore sujet à interrogation de la part des chercheurs en économie (viabilité)
- Des entreprises sur ce modèle ont subi des échecs
- D'autres sont florissantes...
- Mais peut-on parler d'UN modèle ?



Les avantages de l'Open Source

- Pour l'utilisateur
 - Liberté de choix, ouverture des formats
 - Pérénnité : vous avez le source...
 - Coût
 - Technologie ouverte, possibilité d'adaptation illimitée
- Pour l'éditeur de logiciels
 - Diffusion rapide, levier marketing
 - Contribution de la communauté au développement
 - Utilisation de briques existantes



Modèle(s) économique(s)

- Quelques traits généraux
 - La liberté du logiciel conduit à un abaissement des barrières (technologiques) à l'entrée
 - Corollaire : difficulté de vivre de la vente d'un produit
 - Modèle => rémunération sur les services
 - Rééquilibrage des forces entre le client (co-créateur) et l'entreprise
 - Importance des communautés
- Douze modèles économiques, 4 familles :
 - Les stratégies de produit
 - Les stratégies de service
 - La libération des logiciels non stratégiques
 - La vente de produits dérivés



Les stratégies de produit

- Le « produit vedette » : produit au centre de la stratégie d'entreprise => rémunération par des services dédiés - importance de la marque (barrière commerciale) - exemples : Jboss, Zope, Jabber, Apache
- Intégrer des composants: valeur ajoutée dans intégration, tests, certifications, support.. - exemple : les distributions Linux, RH, Suse, Ubuntu,....
- La double licence : copyleft basé sur le copyright => versions copyleftée et propriétaire - rémunération sur la vente de licences propriétaires - exemples : MySQL, TrollTech (Qt)
- La licence chronodégradable : permettre la perception d'une « rente » pour rentabiliser l'investissement - principe : t =sortie de la version commerciale propriétaire, $t+dt$ =publication de son code - exemple : AFPL Ghostscript & GNU - Ghostscript (GPL) (dt = environ 1 an)



Stratégie de produit..(suite)

- Le «dynamitage» : Ouverture du code à des fins de diffusion
 - Parallèle avec la politique de la terre brûlée : tuer la concurrence en la privant d'un revenu - exemple : Open Office : perte de 20% de parts de marché de MS Office
 - Déplacer la lutte vers les services (ouverture du code => marché encore concurrentiel)
- La « vitrine » personnelle : un logiciel libre peut être une excellente carte de visite - exemple : Linus Torvalds et Linux, Guido van Rossum et Python, etc



Les stratégies de service

- Le sur-mesure - exemple : ERP5
- Coopétition et édition mutualiste - exemple : IDX-PKI par IdealX
- Les services dérivés – Intégration, formation, etc....



La libération des logiciels non stratégiques

- L'ouverture des gadgets : ouverture de tout ce qui n'est pas directement générateur de revenus (pilotes, etc) - exemple : Hewlett-Packard (pilotes d'imprimantes), Apple (base Darwin)
- L'ouverture de l'infrastructure : un site d'information vend de l'information, pas son infrastructure – exemple : LinuxFr basé sur Templeet, Wanadoo (FT) Messenger basé sur Jabber, Sun avec Solaris 10 et Java...



La vente de produits dérivés

- Les accessoires
 - Merchandising (OpenStuff.Net), livre (O'Reilly)
- Prolongation du cycle de vie (rémunération sur des activités périphériques)
 - Exemple : ID Software (ouverture de Doom quand déclin et vente de scénarios)

3 – Panorama de l'offre logicielle actuelle (libre ou OS)

Panorama de l'offre logicielle actuelle (libre ou OS)

- Aujourd'hui, plusieurs milliers de logiciels libres ou open source sont disponibles
- Pour pratiquement tous les usages, particuliers ou professionnels
 - (Sauf marchés de niche)
- Tous ne sont pas de même valeur
- De plus en plus d'entreprises utilisent ces logiciels

Panorama : Le poste de travail

- Système d'exploitation : Ubuntu, Fedora, Open Suse, open Solaris, FreeBSD...
- Interface utilisateur : KDE, gnome, E17
- Suite bureautique : OpenOffice, Koffice,...
- Création diagrammes : Dia, Kivio
- PAO : Gimp, Scribus
- Gestion projets: Dot project, Gantt project
- Web & Mail : Firefox, Thunderbird, Zimbra

Panorama : Groupware et dev

- Développement : Java, Kdevelop, GNU,.....
- Développement collaboratif : SVN, CVS
Mercurial,...
- Test : TestNG, Junit, SaloméTMF,..
- Groupware : eGroupware, OpenXchange,
Collaba, OpenGroupware, Zimbra,...

Panorama : Coté serveur

- OS serveurs: Red HAT, Suse, Debian
- Web : LAMP
- Serveur application : Jboss, Tomcat, Glassfish,...
- SGBR : Postgresql, MySql, Firebird, BerkeleyDB

Et les outils métiers :

- CMS : Mambo, Ezpublish, Typo3, Spip,...
- ECM : Alfresco
- File/Print Server : Samba
- CRM : SugarCRM, Vtiger, ...
- ERP : Bravo3, Compiere, ERP5,...
- BI : Spago BI, Talend, Pentaho, Jaspersoft..
- Network Management : Nagios, Ntop, Cacti, NetSnmp

4 - Risques et opportunités du choix de l'Open Source en entreprise

Avantages du modèle

Coûts réduits

pas de licence payante

Plus de services de personnalisation et d'accompagnement du changement !

Personnalisation

code modifiable, solution sur mesure

Indépendance vis à vis de l'éditeur

choix des upgrades, des prestataires

Avantages du modèle

Interopérabilité

emploi de standards ouverts (RFC, norme, format)

Stabilité

correction de bugs, alertes, transparence

Rapidité d'évolution

Sécurité

Qualité du support

forum, mailing liste

Risque du modèle

- **Aspects juridiques**
- **Trop grande variété**
- **Qualité variable, non garantie**
- **Support ? (choix du prestataire (patches, ...))**



Bilan

POUR	CONTRE
<i>Qualité (qualité technique, révision des bogues)</i>	<i>Finition (interfaces moins finies)</i>
<i>Réactivité (mises à jour et corrections fréquentes)</i>	<i>Risque de divergence (projet qui se scinde en plusieurs projets)</i>
<i>Pérennité (garantie par une communauté de développeurs suffisamment grande et active)</i>	<i>Image de marque (l'Open Source garde encore une image de produits « pas sérieux », « de bidouilleurs »)</i>
<i>Coûts (coût d'acquisition souvent gratuit ou réduit)</i>	<i>Manque d'un interlocuteur unique (morcellement de l'offre entre différents interlocuteurs potentiels)</i>
<i>Liberté (Indépendance des choix stratégiques de sociétés commerciales)</i>	<i>Copyleft (non exclusivité de la création logicielle)</i>
<i>Concurrence (prévient l'apparition de monopoles basés sur la fermeture du code source)</i>	
<i>Réutilisabilité (échange et réutilisation de composants)</i>	



Comparaison Evaluation

Propriétaires

Documentation

- Plaquettes commerciales
- Copies écrans, CD démo
- Témoignages clients

Réponse cahier des charges l'information

- Consultant avant-vente
- Filtre par l'éditeur
existantes

Services gratuits AV

- Pay back sur licences
services

Libre

Self service

- Téléchargement
- Tests
- Forums

100% accès à

- Liste des bugs
- 100% fonctions

Licences gratuites

- Pay back sur

Comparaison Apprentissage

Propriétaires

Offre commerciale

Info poussée par éditeur

Complexité

Faire confiance au
vendeur

**Apprentissage après
investissement**

Libre

Tests

Info tirée par le client

Flexibilité

Tester uniquement les
flux critiques

**Apprentissage
progressif**

Comparaison Mise en oeuvre

Propriétaire

Rigidité

Business Templates

Best Practices

Interfaces?

Big Bang

Retour en arrière =€€

Libre

Modèle flexible

Modèle objet

Conçu pour le changement

Ouverture technologique

Améliorations progressives

Go/No Go à tout moment

Comparaison Adaptation

Propriétaire

Achat des sources

Langage propriétaire

**Propriété?
Support?**

Libres

Sources disponibles

Nouvelles technologies

Mutualisation

Comparaison Evolution

Propriétaire

Lois économiques

R&D +/- 15%

Risque disparition

Concentration éditeurs

Visibilité?

Migrations imposées

Libre

Définies par utilisateurs

R&D 100%

Relais pris par intégrateurs

Plans de tests publics

Pas d'obligation de migration

5 – Réussir une implémentation libre ou open source

Critère de choix d'une solution libre ou open source

- Les **méthodes d'évaluation** des logiciels libres
- Basées en partie sur les méthodes déjà existantes pour les logiciels propriétaires
- Certaines intègrent les critères d'évaluation classiques
 - Aspects fonctionnels
 - Aspects techniques
 - Aspects Stratégique du projet ...
- Elles ajoutent les spécificités d'un projet (logiciel) libre
 - Maturité du projet
 - Pérennité

Critère de choix d'une solution libre ou open source

- Quand on évalue un logiciel libre, il existe des questions clés :
 - Quelle est la pérennité du logiciel? Quels sont les risques de Forks ? Comment les anticiper et les gérer ?
 - Quel est le niveau de stabilité auquel s'attendre ? Comment gérer les dysfonctionnements ?
 - Quel est le niveau de support requis et disponible sur le logiciel ?
 - Est-il possible d'influer sur le logiciel (ajout de nouvelles fonctionnalités ou de fonctionnalités spécifiques) ?
 - etc...



Les 4 principales méthodes d'évaluation

Critères	OSMM Cap Gemini	OSMM® Navica	QSOS	OpenBRR
Ancienneté	2003	2004	2004	2005
Auteurs originaux/Sponsors	Cap Gemini	Navicasoft	Atos Origin	Carnegie Mellon West, SpikeSource, O'Reilly, Intel
Licence	Pas libre mais distribution autorisée	Modèles d'évaluation sous Academic Free License	Méthode et évaluations sous licence FDL	Evaluations sous licence Creative Commons
Modèle d'évaluation	Pragmatique	Pragmatique	Pragmatique	Scientifique
<i>Niveaux de détail</i>	2 axes sur 2 niveaux	3 niveaux	3 niveaux ou plus (grille fonctionnelle)	2 niveaux
<i>Critères prédéfinis</i>	Oui	Oui	Oui	Oui
<i>Critères techniques/fonctionnels</i>	Non	Non	Oui	Oui
Modèle de notation	Souple	Souple	Strict	Strict
<i>Echelle de notation</i>	1 à 5	1 à 10	0 à 2	1 à 5
<i>Démarche itérative</i>	Non	Non	Oui	Non
<i>Pondération des critères</i>	Oui	Oui	Oui	Oui
Comparaison	Oui	Non	Oui	Non

Détail d'une méthode : la méthode QSOS

- Méthode de Qualification et de Sélection de logiciel Open Source
- Mise à disposition sous licence libre par Atos Origin.



La méthode QSOS

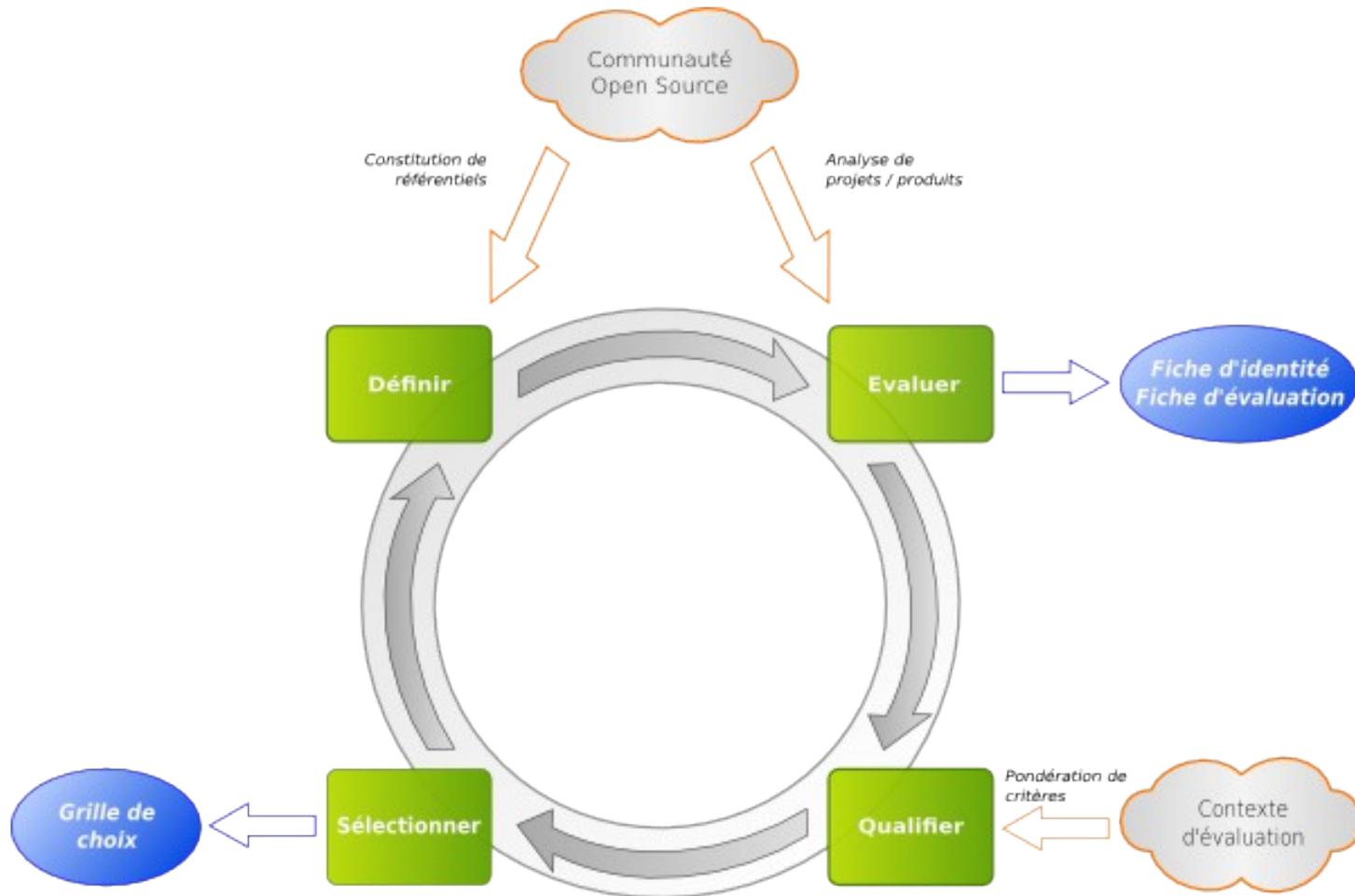
Méthode itérative, en 4 points :

- **Définir** les données de référentiel (types de licences, types de communautés, grilles de couverture fonctionnelle par domaine, ...)
- **Évaluer** les logiciels selon trois axes principaux :
 - couverture fonctionnelle,
 - risques du point de vue de l'entreprise utilisatrice,
 - risques du point de vue du fournisseur de services (expertise, formation, support).

Chaque axe est constitué d'un certains nombre de critères.

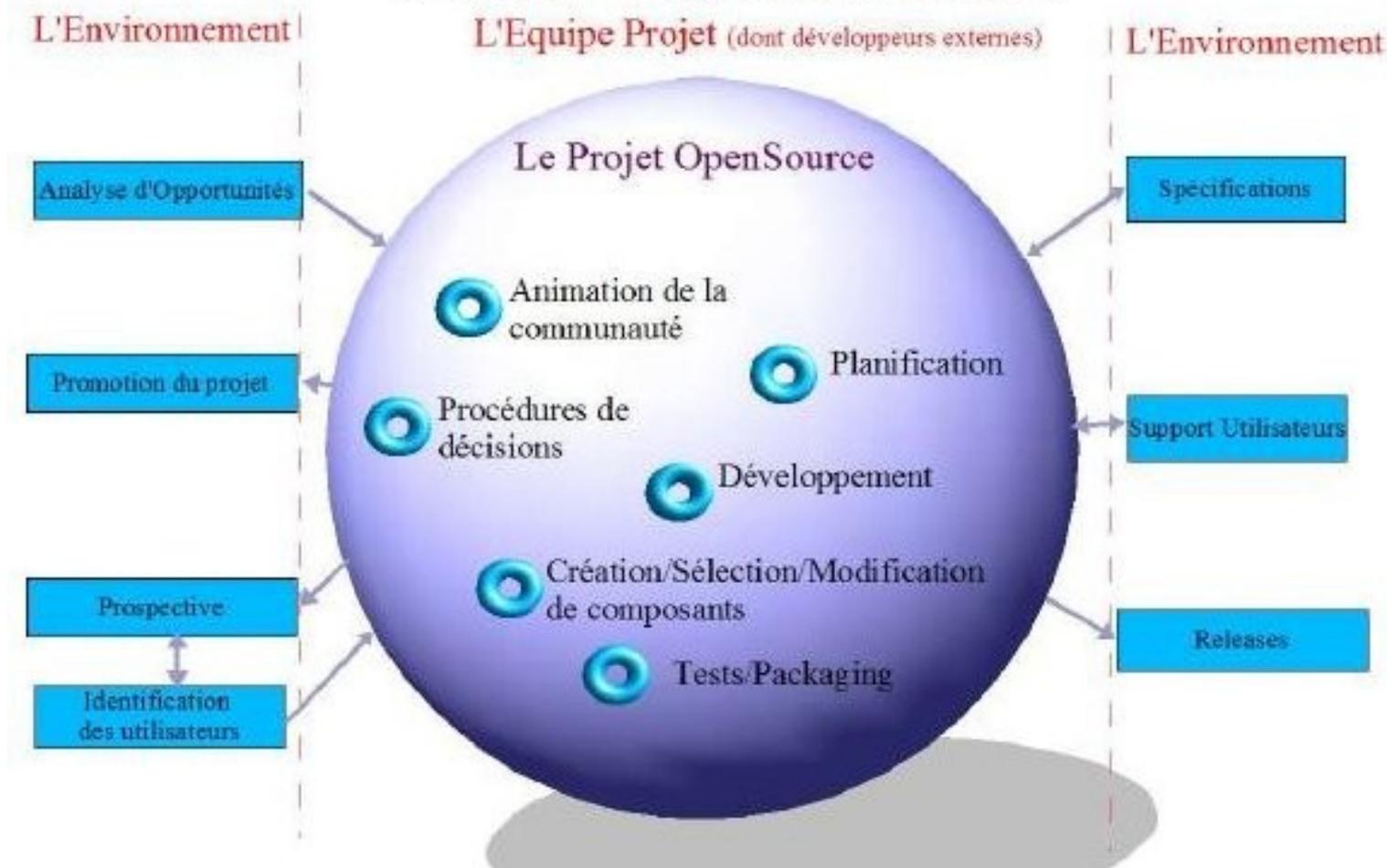
- **Qualifier** le contexte spécifique d'une entreprise (ou d'un utilisateur) en effectuant une pondération des critères précédents.
- **Sélectionner** et comparer les logiciels répondant aux besoins.

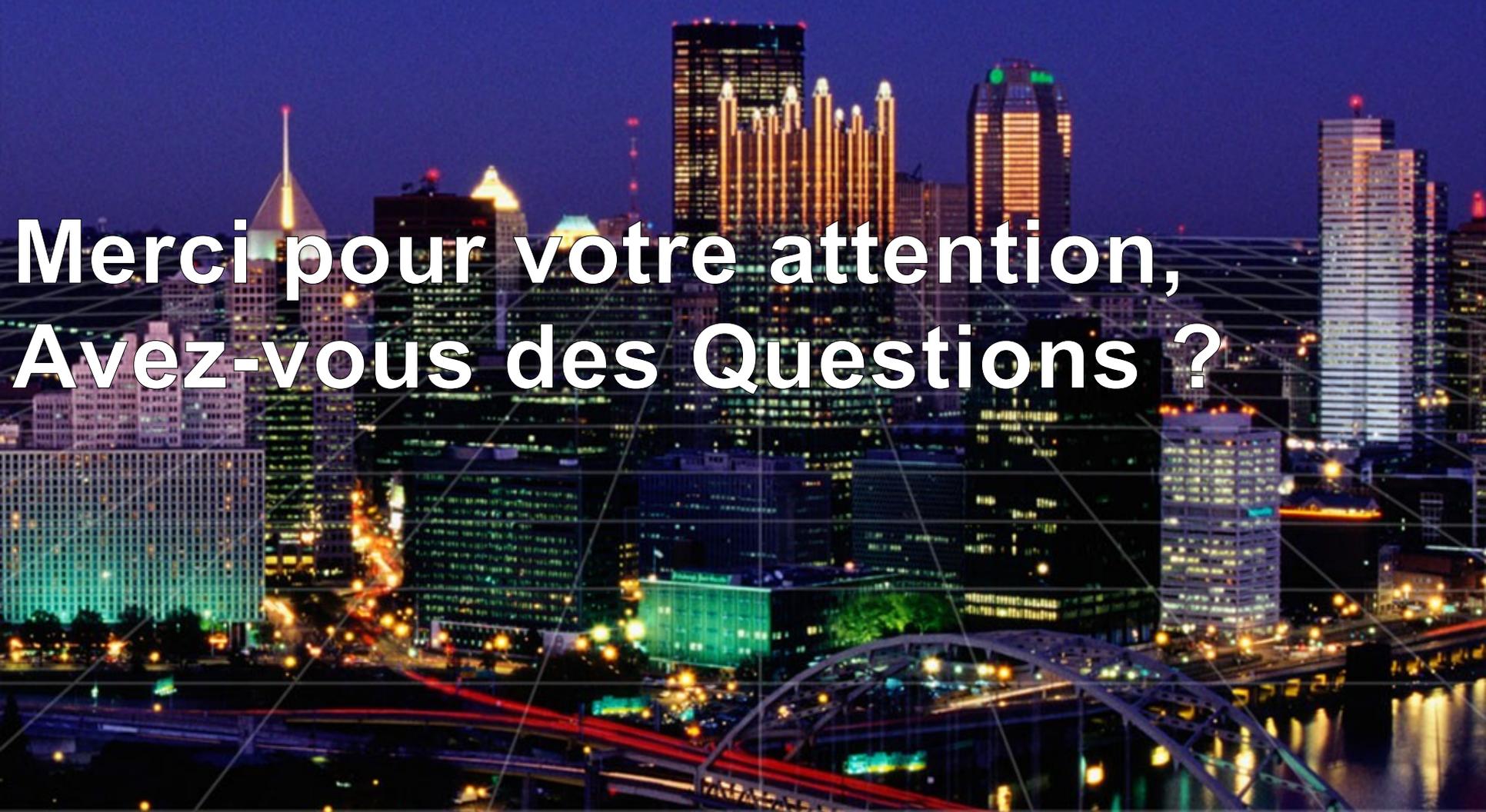
La méthode QSOS



Bulle structurelle

La « Bulle -Projet » OpenSource





Merci pour votre attention,
Avez-vous des Questions ?



JANUA

OPEN SOURCE & OPEN STANDARDS

Pascal FLAMAND
pflamand@janua.fr
0 950 260 370