



SOA Open Source

Intégration des services et business process dans une architecture SOA Open Source

Bruno Georges
JBoss, a Division of Red Hat

Agenda

- **Cas d'étude**
- **Contexte métier**
- **Les bénéfices Open Source**

- **Partager notre expérience et nos défis**
 - Implémentation d'une architecture SOA Open Source
 - Migration vers Jboss

- **Feuille de route vers une architecture SOA Open Source**
 - Communiquer la stratégie
 - Modèles d'intégration
 - Gestion de processus métiers avec jBPM
 - Produits, standards et outils
 - Mesurer le succès et le retour sur investissement (ROI)

- **Etapas suivantes et futurs défis**

Cas d'étude

- Une des plus grandes sociétés privées mondiales de négoce
- Plus de 120 Milliards de US\$ de CA en 2007
- 2000 employés dans le monde entier
- 50 bureaux dans 40 pays
- Initiative SOA globale
- Modèle de taxe complexe, conformité



Vue Globale de l'Application

■ Client Serveur

- Comptabilité/Finance
- Traffic et opérations physiques

■ Applications Web

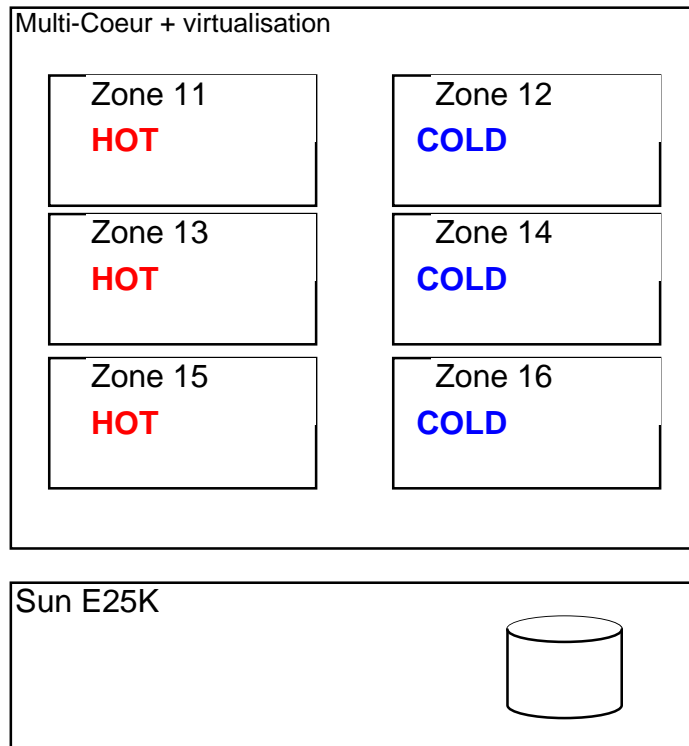
- Gouvernance du SI
- Alimentation des prix
- Applications de négoce et financières

■ Architecture basée sur JEE

- Implémente la logique métier dans les domaines du négoce, de la finance, de la comptabilité, du Forex et de couvertures à terme
- Services d'infrastructure coeur, authentification/autorisations, Audit ...

Infrastructure Physique

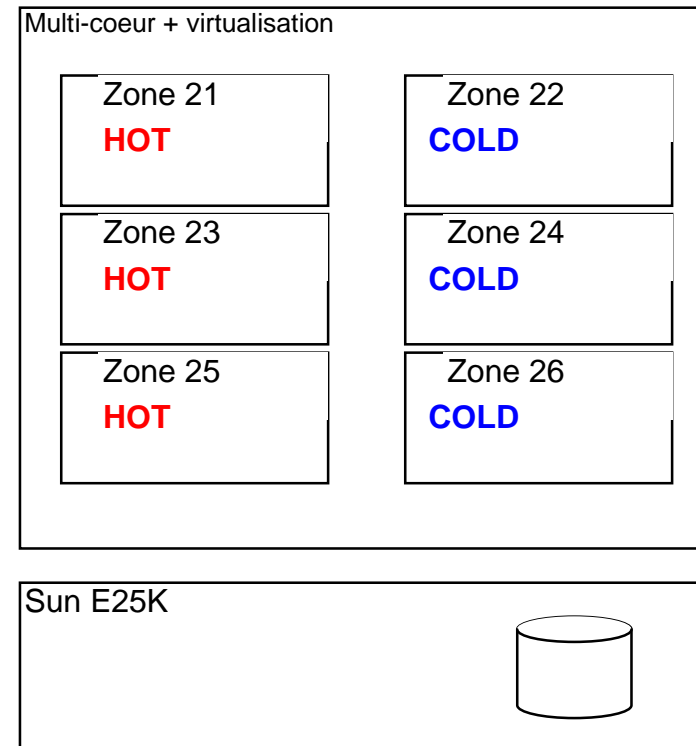
Noeuds primaires



Disques
RAID



Noeuds extérieurs



Disques
RAID

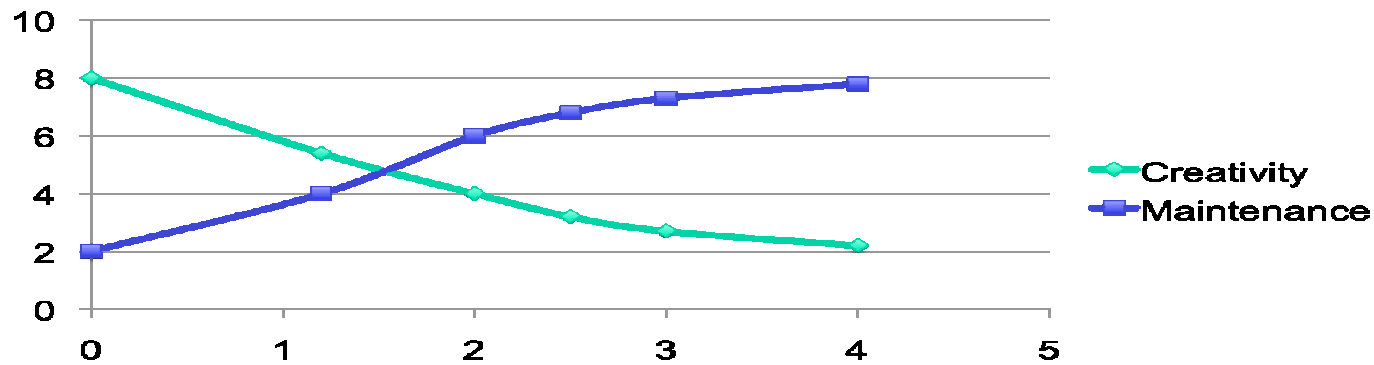


Liens redondants en Fibre Optique - 4GB/sec

JBoss : quels bénéfices ?

■ Processus métiers et intégration de données

Fusion et acquisition – dilemme de l'intégration



■ Workshop SOA

Stratégie globale d'investigation sur les alternatives en open source

■ Contexte Métier

- Licence / Coût total de possession (TCO)
- Etendre SOA aux autres sites, optimiser les coûts
- Time to Market



JBoss : quels bénéfices ?

■ Support

- Répond au besoin de réactivité
- Répond à la nécessité de s'appuyer sur un réseau de partenaires certifiés

■ Développement

- Préférence forte pour l'Open Source
- S'appuie sur des ressources externes plutôt qu'internes
- Simplicité, réduit la dépendance au code spécifique à une infrastructure
- Construit un ensemble commun de connaissances et de bonnes pratiques

Quels défis ?

Du point de vue Métier

- **Migrer la logique métier coeur d'une plateforme JEE vers une autre** sans arrêt de service notable, ni perte de performance, en préservant l'intégrité des données
- **et plus important** : sans affecter les transactions financières, les activités de négoce, les couvertures à terme
- **Accroître la capacité d'intégration pour améliorer l'exécution du métier et réduire les coûts de développement et d'infrastructure**
 - Définition et accord global sur le modèle, les processus métiers, les règles et la logique
 - Définition et publication des points d'intégration des services métiers coeur



Défis techniques

- Maintenance de deux systèmes pendant la migration, fusion des modifications du code
- Gestion des dépendances des services qui sont migrés, considérations supplémentaires si des fonctions spécifiques à un éditeur sont utilisées
- Gain en terme de performance, qualité et stabilité
- Construire une base commune de connaissances pour les développeurs et les ingénieurs. Permet d'accroître la réactivité et d'améliorer le niveau de service



SOA Open Source

Comment atteindre cet objectif ?



Feuille de route

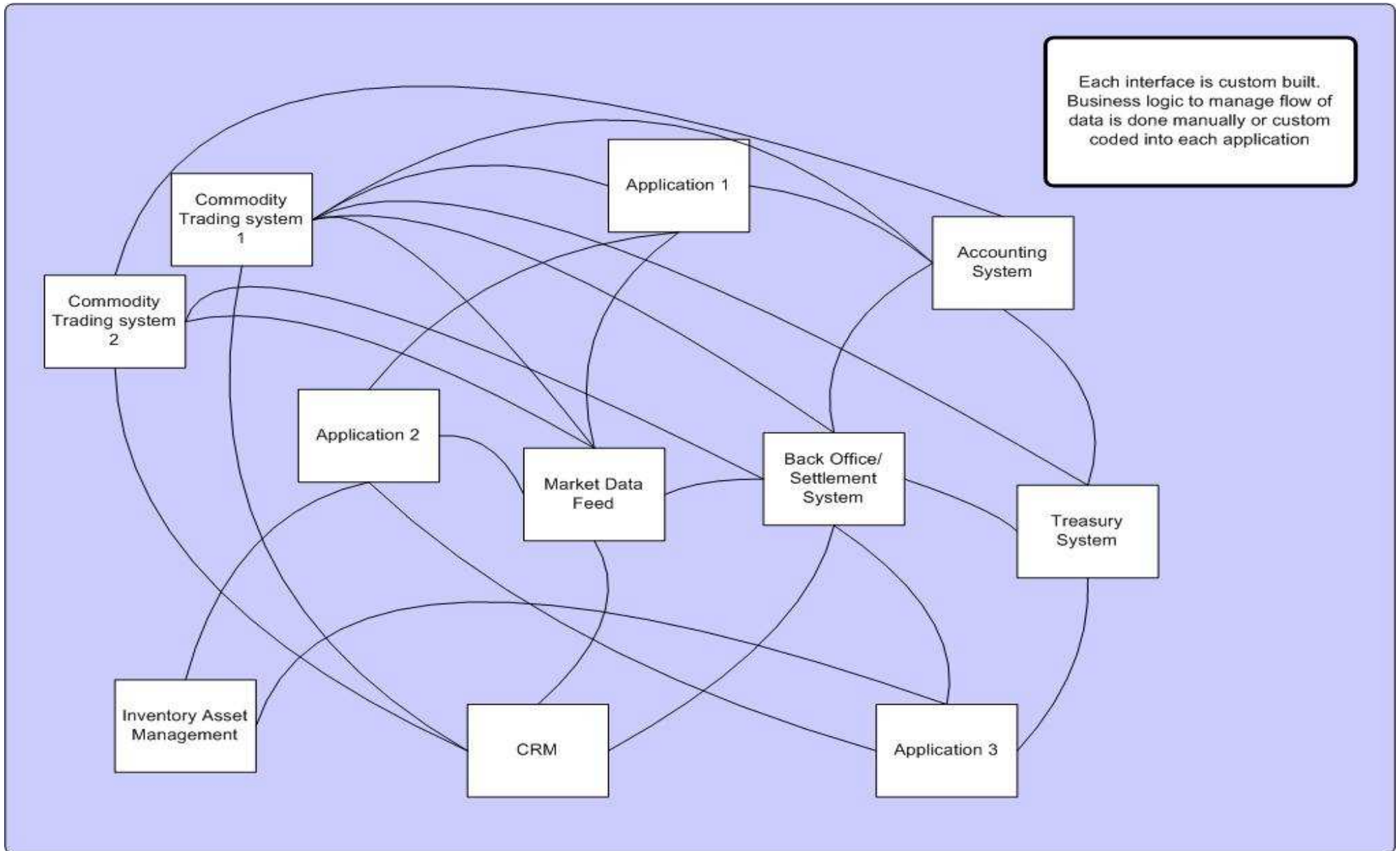
- **SOA**
 - Définir les services, les points d'accès
 - Définir les schéma
 - Définir l'orchestration des flux des processus métiers
 - Définir l'infrastructure SOA => **Open Source**
- **Approcher JBoss**
- **Sélectionner les bons partenaires**
 - Développer de nouvelles relations
 - Evaluer les compétences locales
- **Démarrage du projet de migration**
 - Conduire une évaluation initiale
 - Constituer une équipe
 - Préparer un planning et un cadre
 - Décider : Go / No Go
 - Migrer
 - En parallèle, développer l'expertise en interne



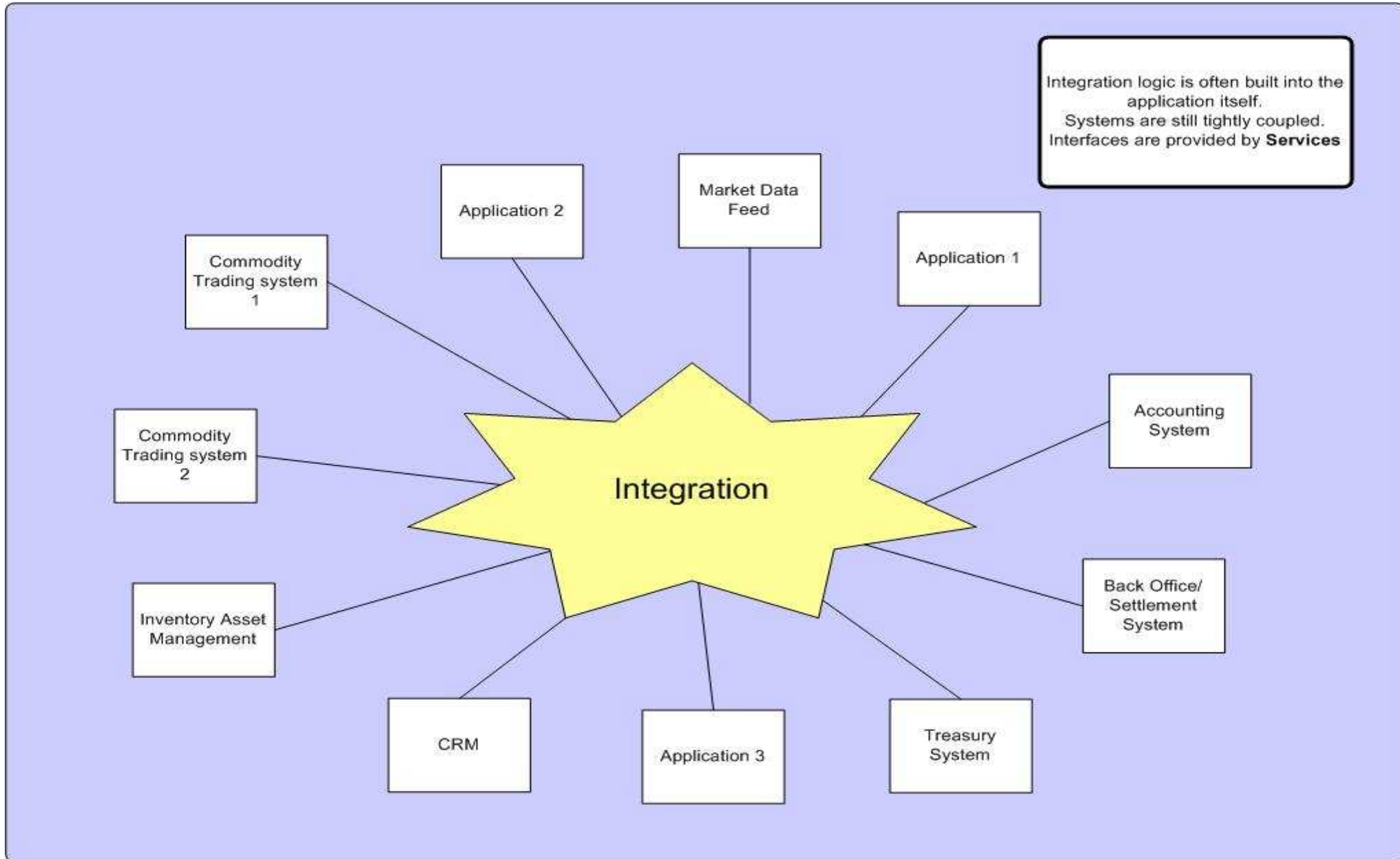
SOA Open Source

Par où avons-nous commencé ?

Basculer vers la notion de Services



Basculer vers la notion de Services

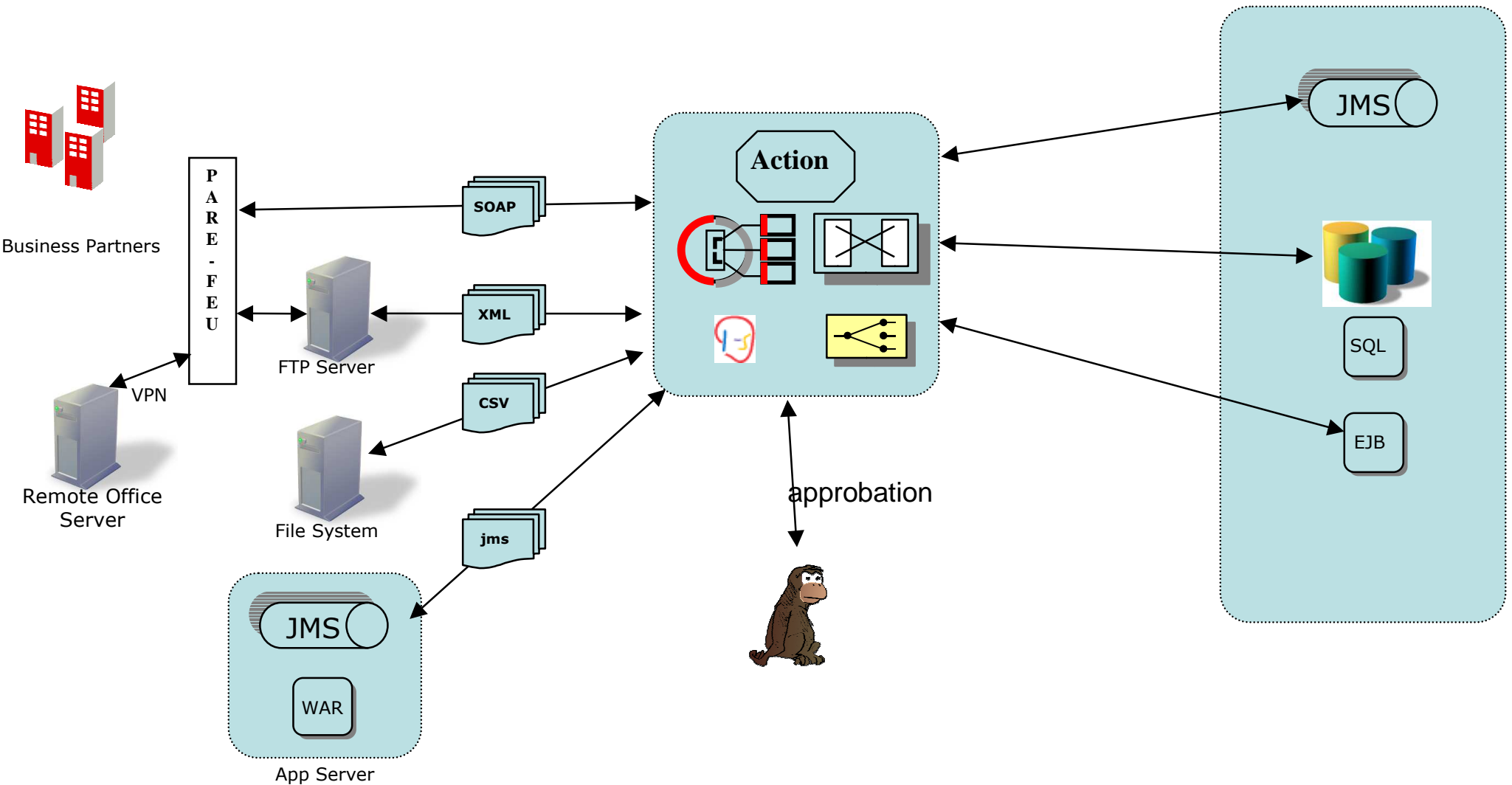


Cas d'utilisation – Vue d'ensemble

Partners - EIS

Integration

Services





Rôles & Responsabilités

- **Architecte d'intégration métier**

- **Développeur d'intégration technique**

- **Rôles supplémentaires pour le succès opérationnel et de déploiement :**
 - Gestion de Projet
 - Build Management et déploiement
 - Tests
 - Opérations
 - Support



Intégration de processus

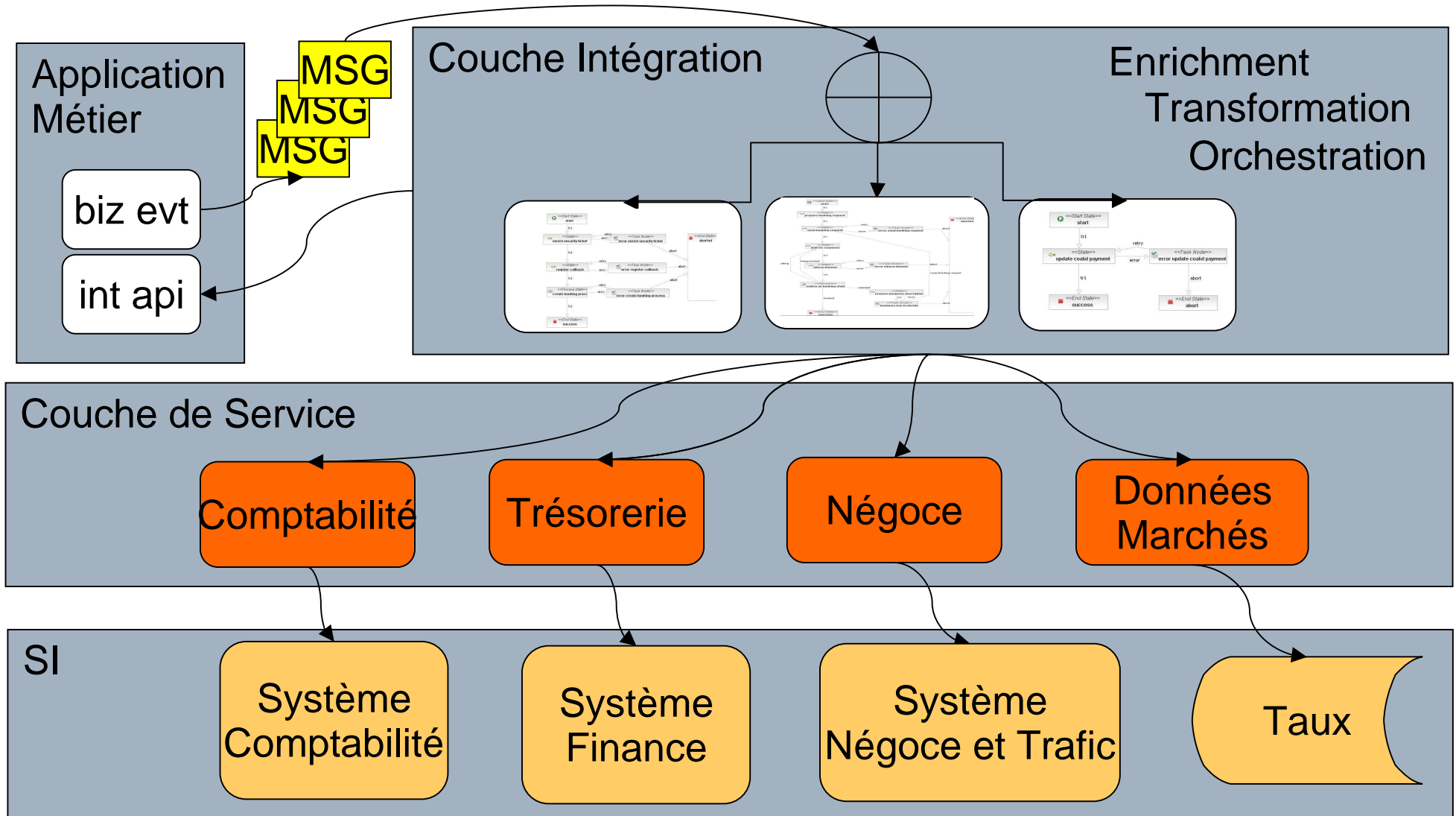
Comment avons-nous fait ?



Pré-requis pour l'intégration

- Découpler les Services des applications qui les consomment
Nous ne contrôlons pas toujours l'infrastructure sur laquelle les applications s'exécutent
- Transformation rapide et orchestration
- Transactionnel, Résilient et Robuste
- Infrastructure de réseau commune
- Standards commun comme XML et SOAP permettant de prendre en charge des processus orientés documents, facilite la réutilisation et favorise le couplage faible

Intégration – Comment ?



Couplage faible

- **SOA est un style d'architecture permettant d'atteindre un haut degré de couplage faible**

Un service est une unité de travail effectuée par un fournisseur de service afin de livrer un résultat attendu par un consommateur

- **SOA cache délibérément ce qui est réalisé derrière un point d'appel de service**

Seul le transfert de données structurées entre les différentes parties nous intéresse

- **SOA permet d'exprimer les fonctions métiers en terme de services pouvant être réutilisés et accessibles via une interface standard**

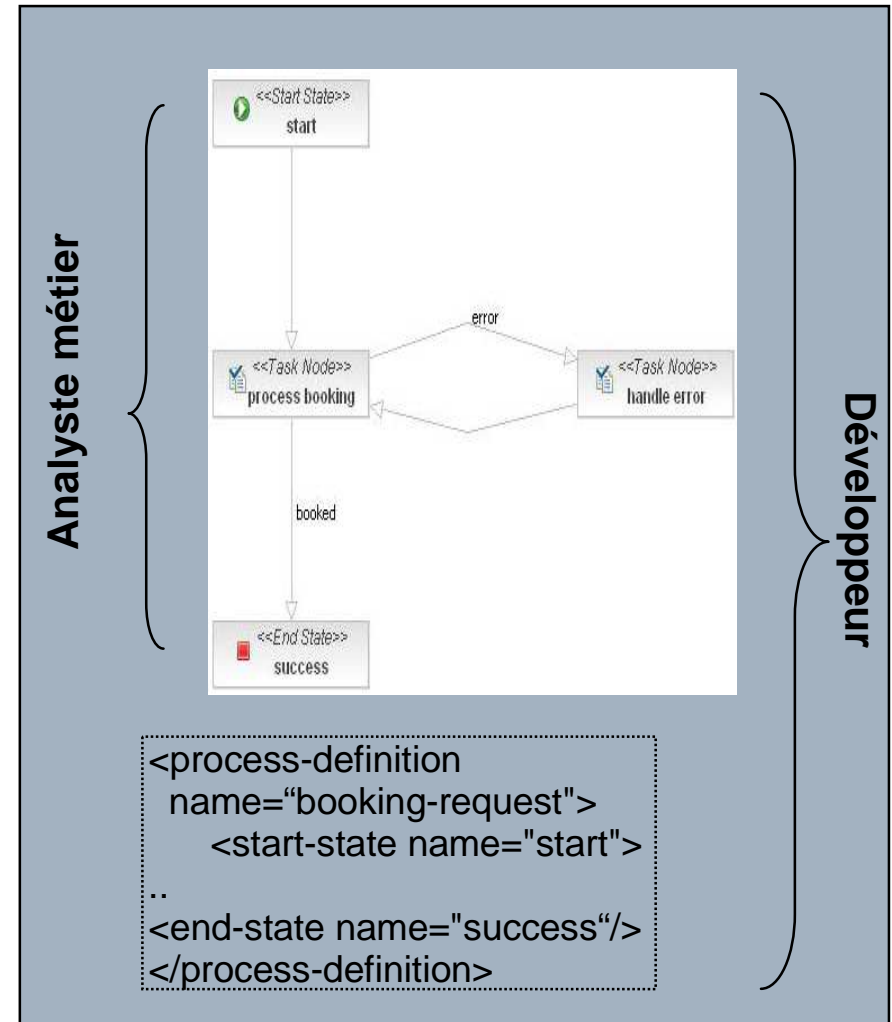
Doivent être accessibles par différentes applications au travers d'une variété de canaux

- **Mais ...**

Il existe différents degrés de couplage et il vous faut choisir celui qui convient à votre cas d'utilisation. Cela pourra être la définition d'interfaces spécifiques pour un seul service ou juste une opération unique

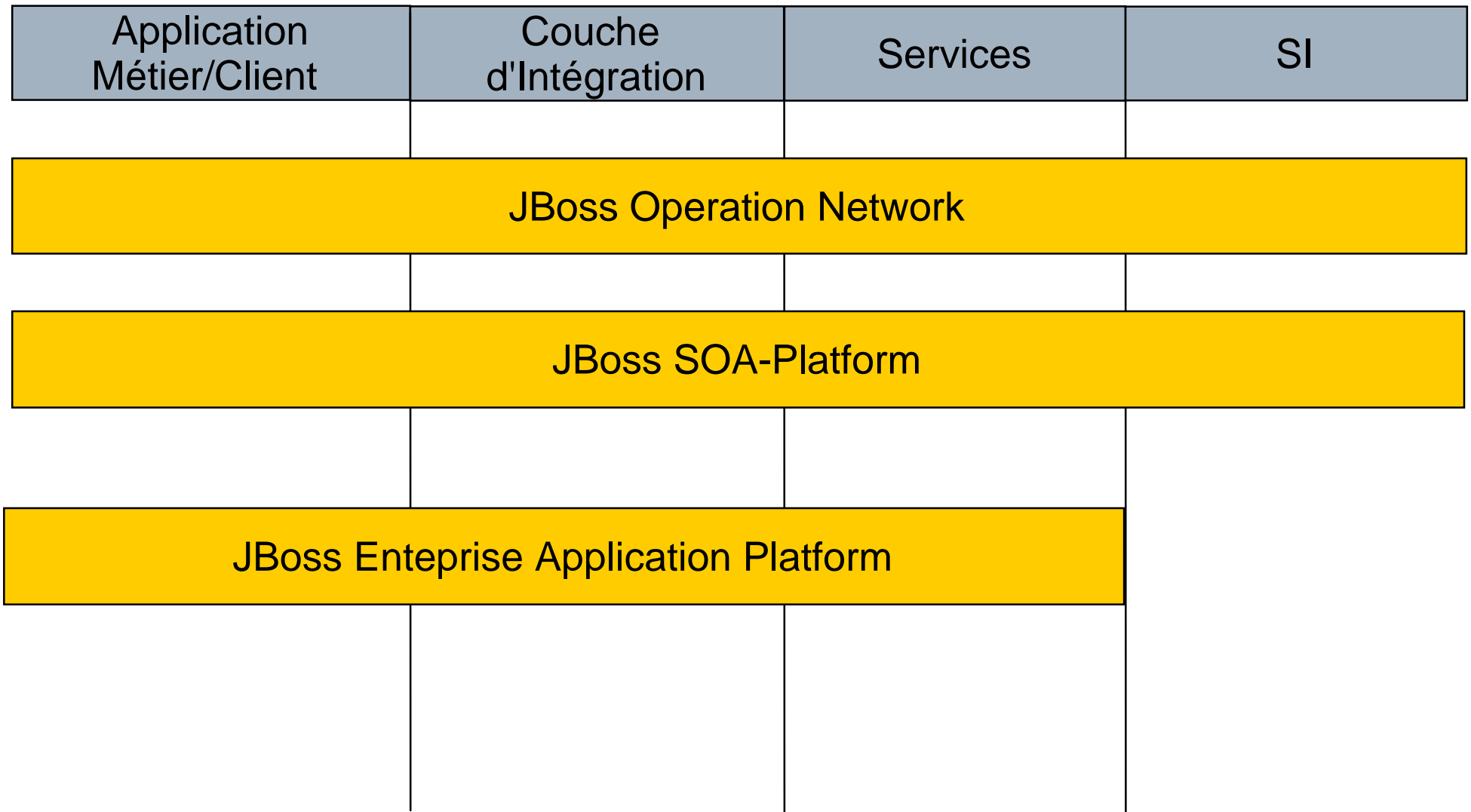
Orchestration de Services

- jBPM pour l'orchestration de processus
- La meilleure solution aussi bien pour les analystes que pour les développeurs
- Cycle de vie de développement avec des itérations rapides
- Processus versionnés
- Communication facilitée entre le développeur et l'analyste métier
- Points d'invocation, intégration de définitions de processus existant au sein de JBoss ESB.
- Indépendant du mode de transport





Composants JBoss





Mesure du succès et Retour sur investissement

- Investissements dans la qualité et l'équipe de développement, PAS dans les licences
- Les changements ne sont pas limités au binaires, ils se reflètent à tous les niveaux de l'organisation du SI
- Cycles de livraisons plus rapides
 - Moins de temps passé dans l'intégration, les test, le code d'infrastructure
 - Réduit les temps de réponse
- Economie totale sur 3 ans
 - > 3,5 M US\$**



Prochaines étapes

■ Architecture

- Adoption des Standards pour une SOA intéroperable
 - => Livraison de message / WS-Addressing / WS-Security
 - => Registre de Service - UDDI
- Investir dans une infrastructure SOA. JBoss ESB

■ Développement

- Outillage de création de Service, EJB3, Annotations, JSR181
- Cycle de vie de Service (modelage, assemblage, déploiement, gestion)

■ Gouvernance

- Outils au runtime, Business Activity Monitoring (BAM)
- Contrats de Service, Référentiel de Méta-données

JBoss Enterprise Middleware



JBoss Developer Studio

- Eclipse IDE
- Integrated Tooling
- Runtime Platform

Fully Integrated Developer Environment

Portal Platform

- JBoss Portal
- Application Platform

Content Aggregation, Presentation and Personalization

Application Platform

- JBoss Seam
- Hibernate
- JBoss Application Server

Embedded Tomact, Clustering, Cache, Messaging, Transactions

Application Presentation, Services Hosting, and Data Persistence

SOA Platform

- JBoss jBPM
- JBoss Rules
- JBoss ESB
- Application Platform

Transformation, Routing, Registry, Repository

Service Integration & Orchestration, Business Process Automation, Rules Definition, & Event Management

MetaMatrix Designer

MetaMatrix Data Services Platform

- MetaMatrix Enterprise Server

Data Integration, Data Service Federation, Data Abstraction & Management

JBoss Operations Network

Administration, Management, and Monitoring

Enterprise Platforms Integrated distribution with a single patch & update cycle

Enterprise Frameworks Modular productivity tools that also work on other vendors application servers

Component major component or set of integrated components



Questions?

bruno.georges@redhat.com