

Réalisation d'une application de Merchandising

**Mémoire,
en vue d'obtenir
le diplôme d'ingénieur CNAM
en informatique,
soutenu le jeudi 6 décembre 2007**

Laurent Dongé

laurent_donge@yahoo.fr

Plan

■ Introduction

■ Projet

↳ Objectifs

↳ Gestion de projet

■ Intégration dans le SI

↳ ETL et EAI

■ Application WEB

↳ Organisation

↳ Outils

■ Architecture

↳ Solution Apache
Tomcat

■ Test de charge

↳ Objectifs

↳ Comparatif des outils

↳ Démarche

■ Bilan

■ Conclusion

Introduction

- **Introduction**
- **Projet**
- **Intégration dans le SI**
- **Application WEB**
- **Architecture**
- **Test de charge**
- **Bilan**
- **Conclusion**

■ Besoin Distributeur

↳ Optimisation de la présentation de l'offre produit en magasin

■ Meilleurs outils (élaboration, diffusion, reporting)

↳ Propriétaires

↳ Spécifiques

■ 2 problématiques

↳ Intégration

↳ Application WEB

- Introduction

- **Projet**

- Objectifs
- Gestion de projet

- Intégration dans le SI

- Application WEB

- Architecture

- Test de charge

- Bilan

- Conclusion

- Objectifs

- ↳ Renouvellement de l'application

- ↳ Augmentation de la productivité

- ↳ Diffusion WEB

- Acteurs (700 déclarés, 10 000 potentiels)

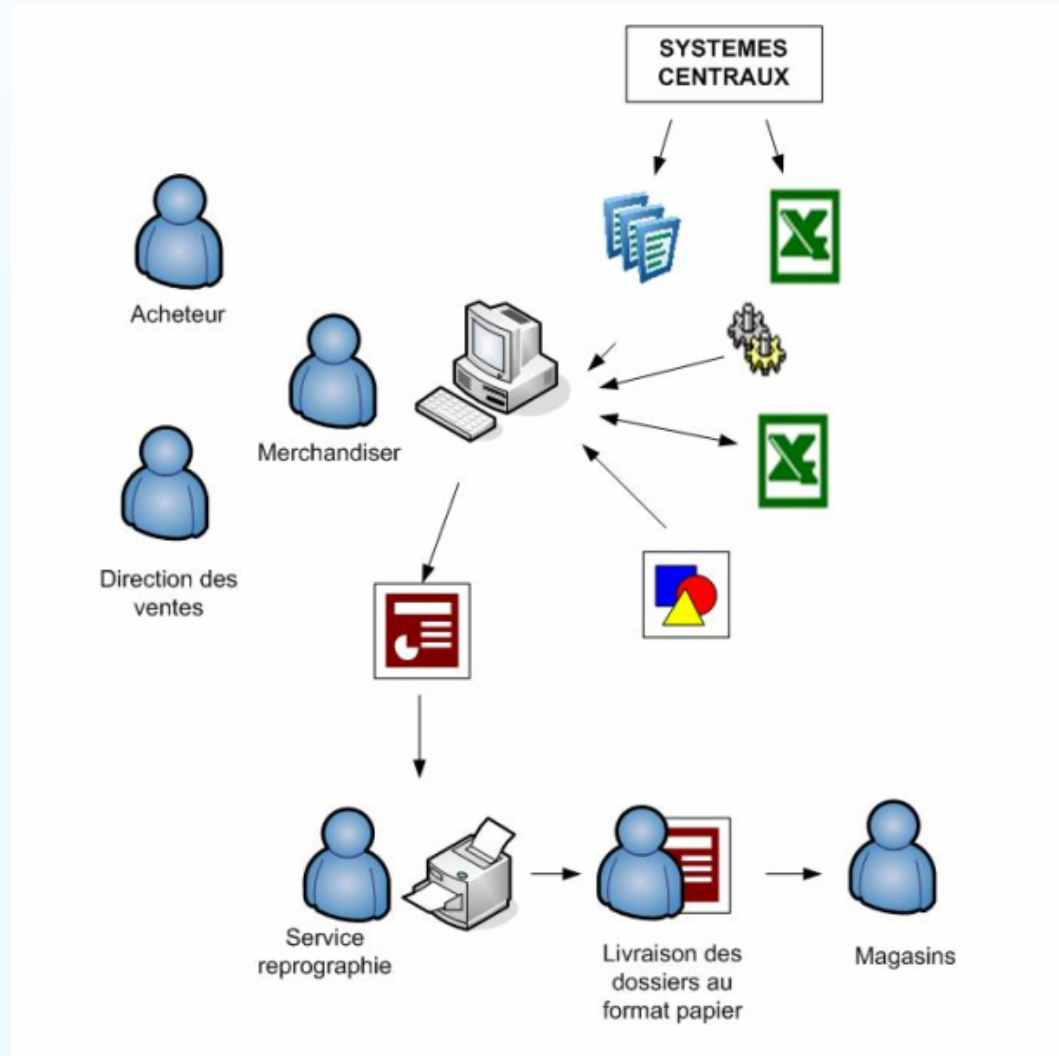
- Le merchandising

Ensemble des techniques visant à optimiser la répartition et la présentation des produits en magasin afin d'améliorer la rentabilité et le trafic et de contribuer à l'expression de l'image de l'enseigne.

- Finalité : des dossiers de préconisation

L'existant

- Introduction
- **Projet**
 - Objectifs
 - Gestion de projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
- Conclusion



Le constat

- Introduction
- **Projet**
 - Objectifs
 - Gestion de projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
- Conclusion

■ Axes d'optimisation

↳ Automatisation, centralisation, interface simple, abonnement image, diffusion WEB ...

■ Gains attendus

↳ Temps, argent et interactivité

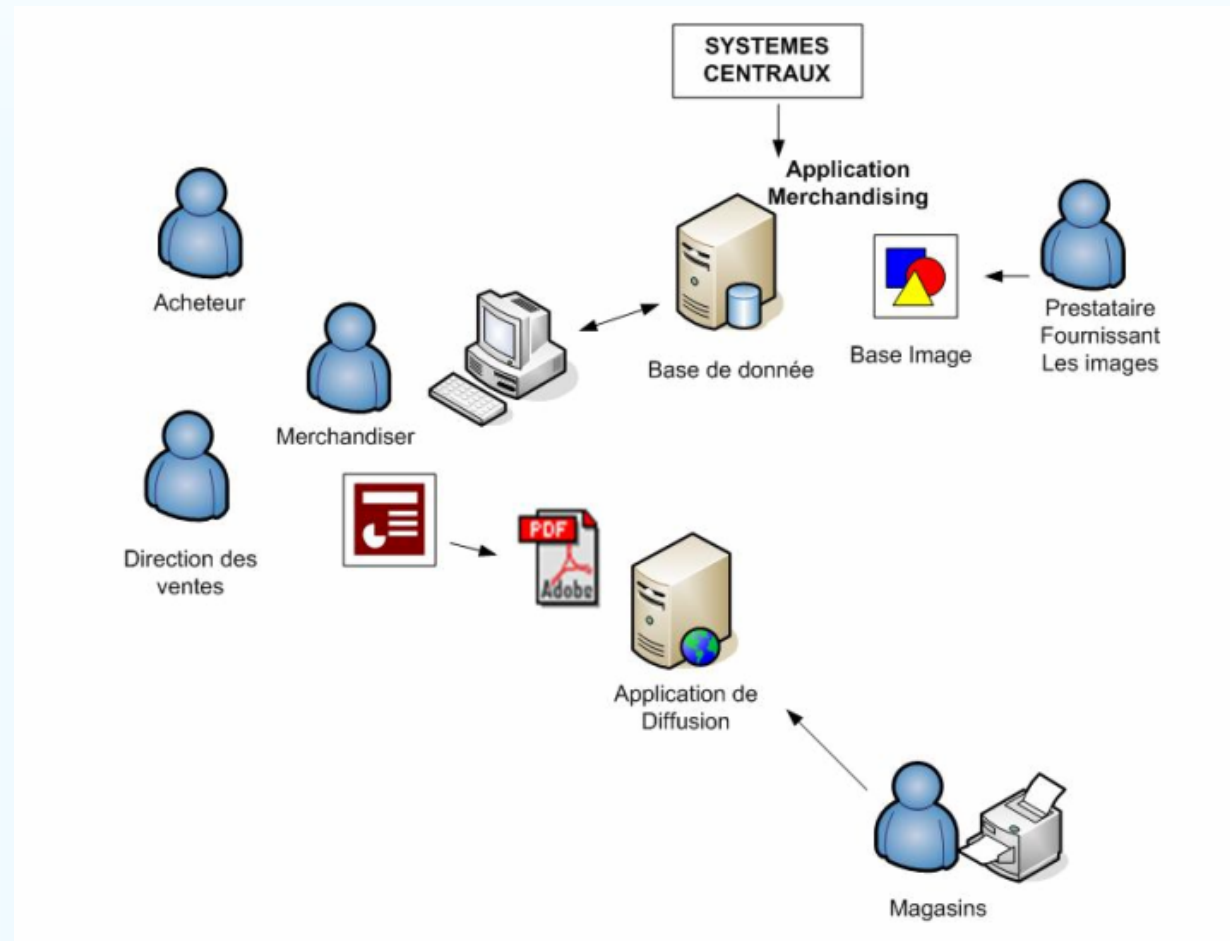
■ Contraintes

↳ SI hétérogène, base image importante, choix DSI, moyens restreints

La cible

1 appli propriétaire + 1 appli spécifique Métier

- Introduction
- **Projet**
 - Objectifs
 - Gestion de projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
- Conclusion



Gestion de projet

- Introduction
- **Projet**
 - Objectifs
 - **Gestion de projet**
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
- Conclusion

- Inspiré d'XP et de 2TUP
- Utilisant Merise et UML
- Reposant sur une équipe organisée
- Recherchant le compromis entre les délais, les coûts, les risques et la qualité: périmètre fonctionnel
- Pragmatique (rentabilité, réutilisation, adaptation)
- Privilégiant l'humain et un logiciel fonctionnant
- Recherchant la simplicité

Le Système d'Information SI

- Introduction
- Projet
- **Intégration dans le**
SI
 - EAI et ETL
- Application WEB
- Architecture
- Test de charge
- Bilan
- Conclusion

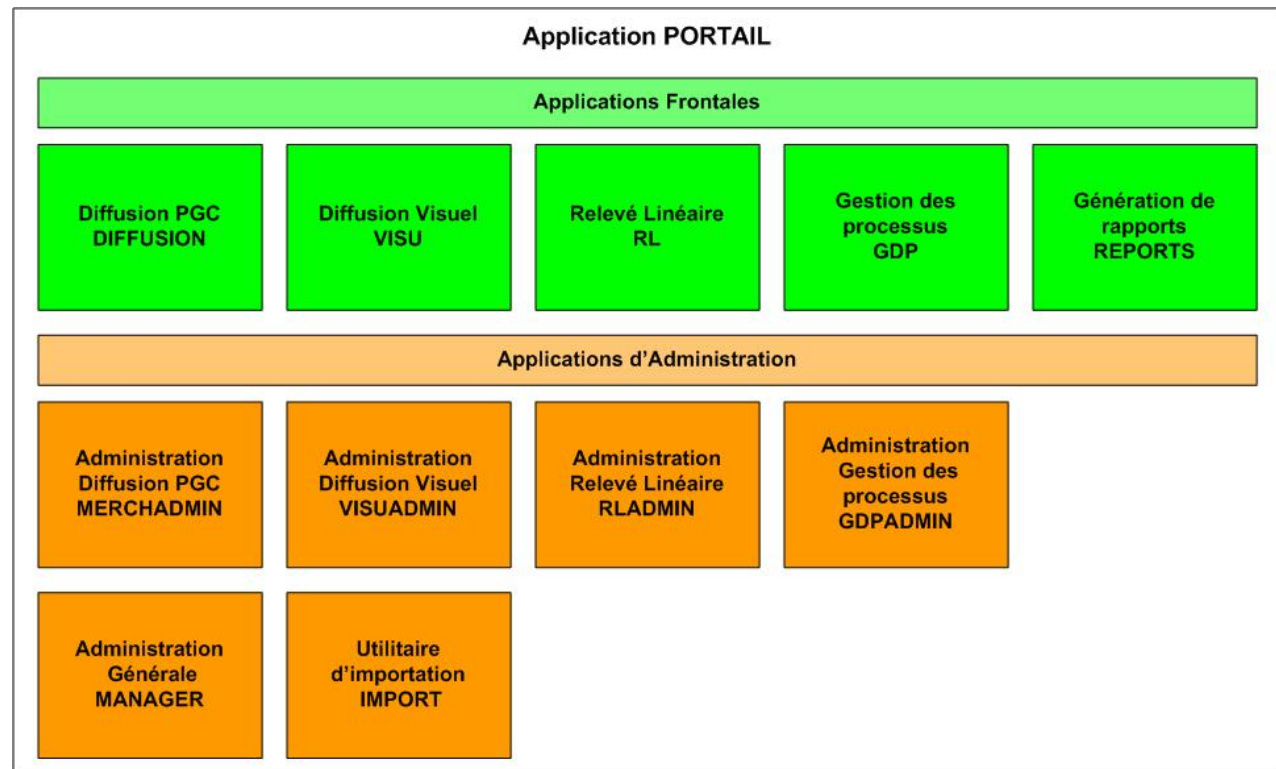
- Besoins d'informations (produits, ventes, magasins)
- SI hétérogène (technologies)
- Pas directement exploitables (retraitements)
- Insuffisantes
 - ↳ Écrans de saisie
 - ↳ Prestation extérieure (images, dimensions)

- Introduction
 - Projet
 - Intégration dans le SI
 - EAI et ETL
 - Application WEB
 - Architecture
 - Test de charge
 - Bilan
 - Conclusion
- Problématiques (temps réel, batch)
 - Comparaison (métier, données)
 - EAI non retenu (coût, lourdeur)
 - Traitements
 - ↳ ETL GENIO
 - ↳ Agents locaux (JAVA)

Architecture de l'application

- Introduction
- Projet
- Intégration dans le SI
- **Application WEB**
 - Organisation
 - Outils
- Architecture
- Test de charge
- Bilan
- Conclusion

Une application modulaire



reposant sur J2EE, RoR et BIRT.

Implémentation J2EE

- Introduction
- Projet
- Intégration dans le SI
- **Application WEB**
 - Organisation
 - Outils
- Architecture
- Test de charge
- Bilan
- Conclusion

■ Plateforme

- ↳ Standard
- ↳ Stable, fiable
- ↳ Indépendance de l'OS
- ↳ Base générique offrant des fonctionnalités.

■ Services

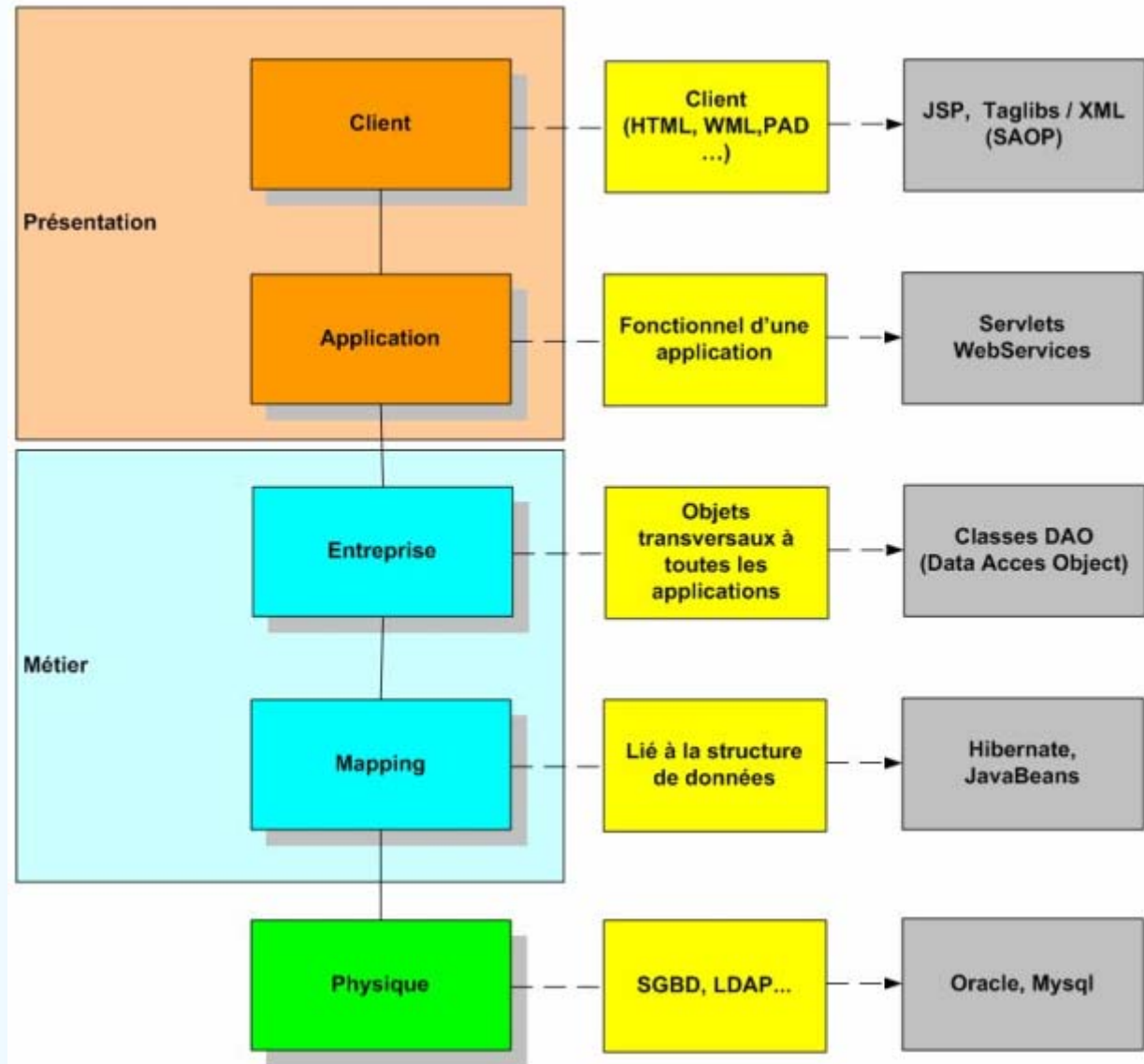
- ↳ Infrastructure (JDBC, JNDI)
- ↳ Communication (Web Services, JavaMail)

■ Repose sur des composants

- ↳ WEB (JSP, servlets)
- ↳ Métier (EJB)

Architecture basique

- Introduction
- Projet
- Intégration dans le SI
- **Application WEB**
 - Organisation
 - Outils
- Architecture
- Test de charge
- Bilan
- Conclusion



Les outils

- Introduction
- Projet
- Intégration dans le SI
- **Application WEB**
 - Organisation
 - **Outils**
- Architecture
- Test de charge
- Bilan
- Conclusion

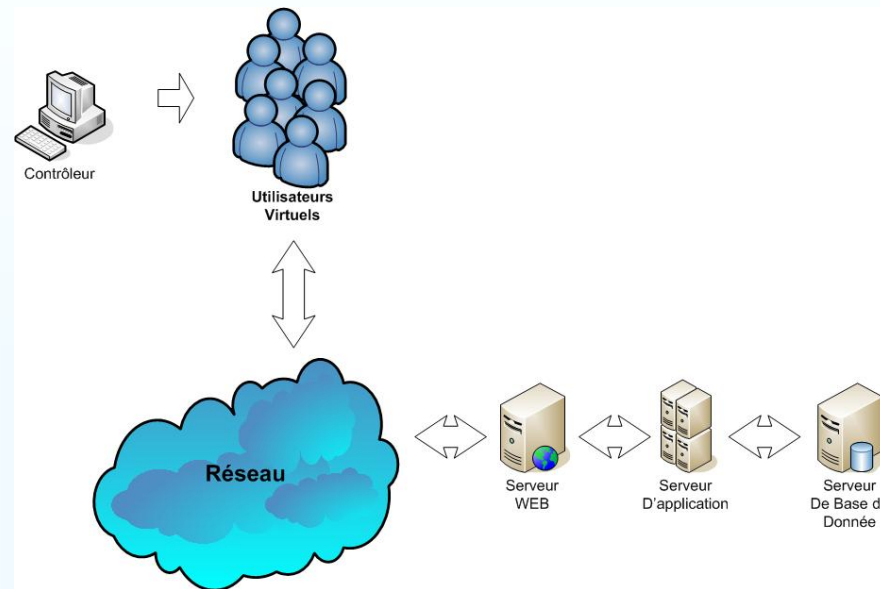
- Framework (Struts, Hibernate)
- Design Patterns (MVC, DAO)
- IDE (Eclipse)
 - ↳ Modules :
Sysdeo, Lombok et Hibernate
 - ↳ Autres outils :
Ant, CVS, Junit, log4j, Tiles
- Ruby On Rails (entre PHP et J2EE, convention, MVC)
- BIRT (Actuate)

Une architecture robuste et adaptable

- Introduction
 - Projet
 - Intégration dans le SI
 - Application WEB
 - **Architecture**
 - **Solution**
 - Apache Tomcat**
 - Test de charge
 - Bilan
 - Conclusion
- Cohabitation de serveurs d'applications (Tomcat 5 et 6, Mongrel)
 - Répartition de charge et site partitioning (apache)
 - Gestion des sessions (affinité et cookies, DB)
 - Base de données (Oracle)
 - Tolérance aux pannes (pas HA)

Objectifs

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
 - Objectifs
 - Comparatif des outils
 - Démarche
- Bilan
- Conclusion



- Anticiper les problèmes
- Fiabiliser et stabiliser l'application
- Valider la plateforme
- Tester la résistance de la plateforme
- Offrir une bonne qualité de service

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- **Test de charge**
 - Objectifs
 - **Comparatif des outils**
 - Démarche
- Bilan
- Conclusion

- Nombreux outils (Opensta, LoadRunner, Webload)
- Gratuit ou payant
- Opensta
 - ↳ Protocole (HTTP,HTTPS)
 - ↳ Actions manuelles
 - ↳ Peu de fonctionnalités
 - ↳ Validation d'architecture

Démarche

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- **Test de charge**
 - Objectifs
 - Comparatif des outils
 - **Démarche**
- bilan
- Conclusion

- Déterminer des objectifs (temps, nb utilisateurs)
- Simuler la charge (modèle)
- Suivre des indicateurs (%CPU, %RAM, bande passante, durée, erreurs)
- Exécuter des tests (vide, charge, tenue)
- Éliminer les goulots d'étranglement

ETL : avantages/inconvénients, un constat positif mais pas idéal

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
 - Intégration
 - Application J2EE
 - Architecture
 - Tests de charge
 - RoR et J2EE
- Conclusion

- Facilité de création, suivi, maintenance des traitements (cadre simple, représentation unique, analyse d'impact)
- Gain de temps (formation, maintenance, complexité visuelle)
- Inconvénients (volumétrie, propriétaire, tarif, version adaptateur)
- Outils complémentaires (agent locaux Java, alternative)

Application J2EE

- Introduction
 - Projet
 - Intégration dans le SI
 - Application WEB
 - Architecture
 - Test de charge
 - Bilan
 - Intégration
 - **Application J2EE**
 - Architecture
 - Tests de charge
 - RoR et J2EE
 - Conclusion
- Framework
 - ↳ Cadrer, simplifier, améliorer la maintenance et gagner du temps (J2EE lourd)
 - Couches : Les possibilités
 - ↳ Client : WEB 2.0, AJAX
 - ↳ Application :
 - Web Services (données référentielles)
 - Spring, JSF et Tapestry
 - ↳ Métier :
 - Conteneur lourd EJB
 - Conteneur léger Spring (transverse)
 - ↳ Physique : MySQL au lieu d'Oracle

Architecture

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
 - Intégration
 - Application
 - J2EE
 - **Architecture**
 - Tests de charge
 - RoR et J2EE
- Conclusion

- Serveur d'application Open Source
 - ↳ Cluster à moindre coût (Tomcat 5)
 - ↳ Architecture évolutive
 - ↳ Réactivité de la communauté
- Cluster pas une solution « magique »
 - ↳ Suppose des développements simples (qualité)
 - ↳ Ne répond pas à tout (LAN, DB ...)
 - ↳ Nécessite une validation (comportement utilisateurs)

Test de charge

- Introduction
 - Projet
 - Intégration dans le SI
 - Application WEB
 - Architecture
 - Test de charge
 - Bilan
 - Intégration
 - Application J2EE
 - Architecture
 - **Tests de charge**
 - RoR et J2EE
 - Conclusion
- Tests du cluster
 - ↳ Serveur dédié Oracle + optimisation SQL
 - Fastidieux, long
 - ↳ Nombreux points de contention possibles
 - OpenSta
 - ↳ Charge parfois peu réaliste (comportement utilisateurs, charge homogène)
 - Processus itératif (évolution)

RoR et J2EE

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- **Bilan**
 - Intégration
 - Application
 - J2EE
 - Architecture
 - Tests de charge
 - **RoR et J2EE**
- Conclusion

- Langage et framework (simple)
- Outils (Eclipse)
- Technologie (Web Service et AJAX)
- Architecture (serveur d'application et cluster)
- Performance (JVM)
- Type de projet
- Solution professionnelle reconnue
- La technologie ne résout pas tous les problèmes (gestion de projet, organisation)

Conclusion (1/3)

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
- Conclusion

- Application propriétaire + spécifique
 - ↳ Meilleure gestion des risques et des dérapages
- ETL (globale, choix DSI, suppression de GENIO)
- Outils spécifiques WEB
 - ↳ J2EE : nombreux atouts
 - ↳ RoR et BIRT : réduction des coûts
- Ruby on Rails
 - ↳ Mise en place rapide de sites structurés
 - ↳ Besoin simple, peu d'utilisateurs (IHM d'administration)
 - ↳ Gain important (80 % de l'application)

Conclusion (2/3)

- Introduction
- Projet
- Intégration dans le SI
- Application WEB
- Architecture
- Test de charge
- Bilan
- Conclusion

- Test fonctionnel : Selenium
 - ↳ Validation fonctionnelle rapide
- Solution Merchandising
 - ↳ Administration à enrichir (gestion de la qualité des données, reporting et suivi)
 - ↳ Évolutions fonctionnelles (plan magasin)
- Objectif
 - ↳ Répondre à un maximum de besoins avec des coûts réduits en se souciant de mettre en place des solutions simples, maintenables et évolutives dans un cadre bien défini dans des délais restreints

Conclusion (3/3)

- Introduction
 - Projet
 - Intégration dans le SI
 - Application WEB
 - Architecture
 - Test de charge
 - Bilan
 - Conclusion
- Chef de Projet Informatique avec un rôle élargi
 - Gestion de projet Agile non répandue chez MONOPRIX (compétitivité, réponse aux besoins, délais)
 - La place de l'expérimentation, de la veille technologique
 - L'acceptation du changement
 - Une démarche intrusive acceptée
 - Une solution Open Source
 - La collaboration plutôt que le contrat (binôme, TQM, gagnant-gagnant)
 - Une adaptation technique, méthodologique et organisationnelle

Questions - Discussion

