

Projet J2EE



Daniel Hagimont

IRIT/ENSEEIH

**2 rue Charles Camichel - BP 7122
31071 TOULOUSE CEDEX 7**

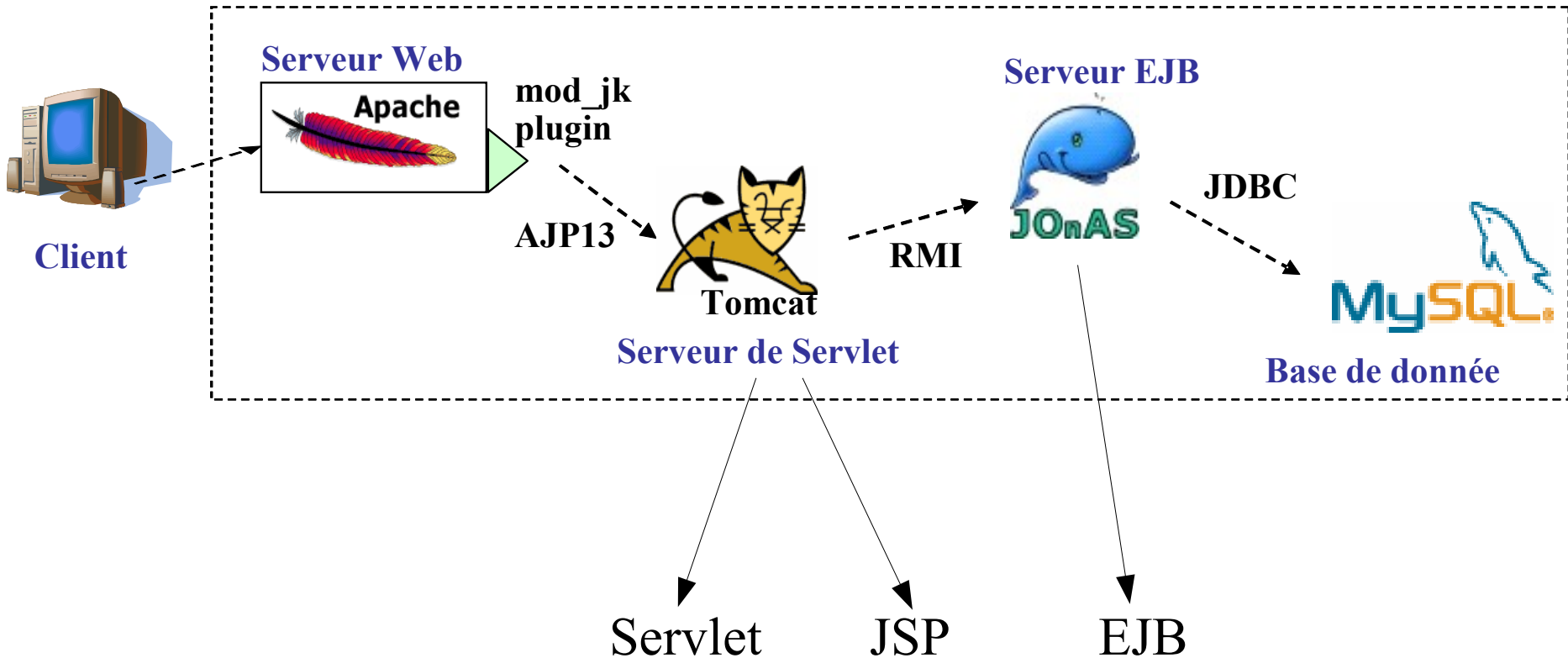
**Daniel.Hagimont@enseeiht.fr
<http://hagimont.perso.enseeiht.fr>**

Projet J2EE

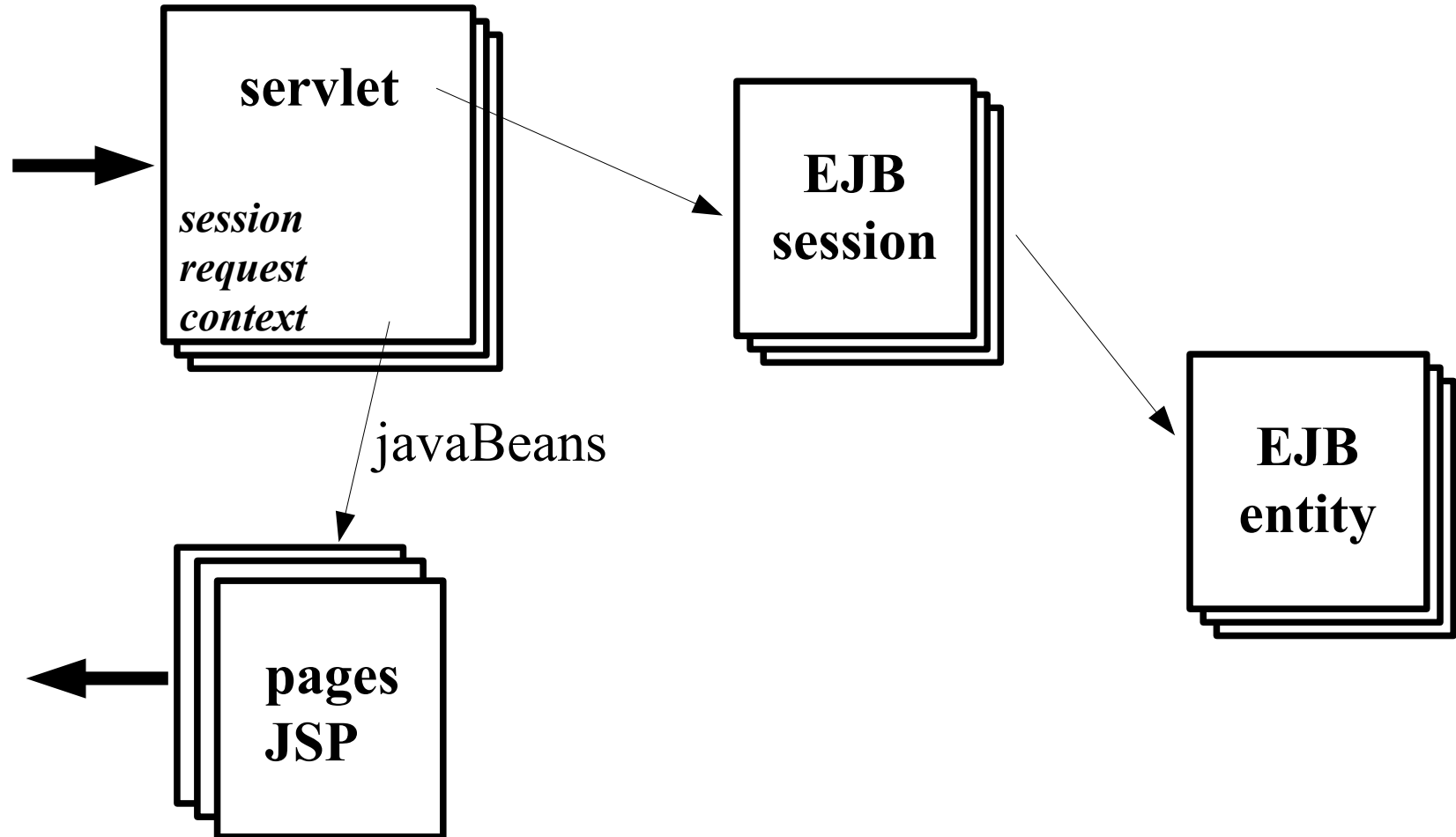


Rappels ...

J2EE



Développement



Technologies additionnelles

■ Struts

- Canevas pour structures des applications MVC
 - JSP pour la vue (avec des tags struts)
 - Classe Java pour le modèle
 - Descripteurs struts (xml) reliant les requêtes/classes/JSP

■ CSS

- Feuille de style pour les pages web
 - Pages web structurées avec des DIV
 - Présentation séparée de la structure

■ AJAX

- Mises à jour de fragments de pages
- Animations

Projet J2EE



Le sujet ...

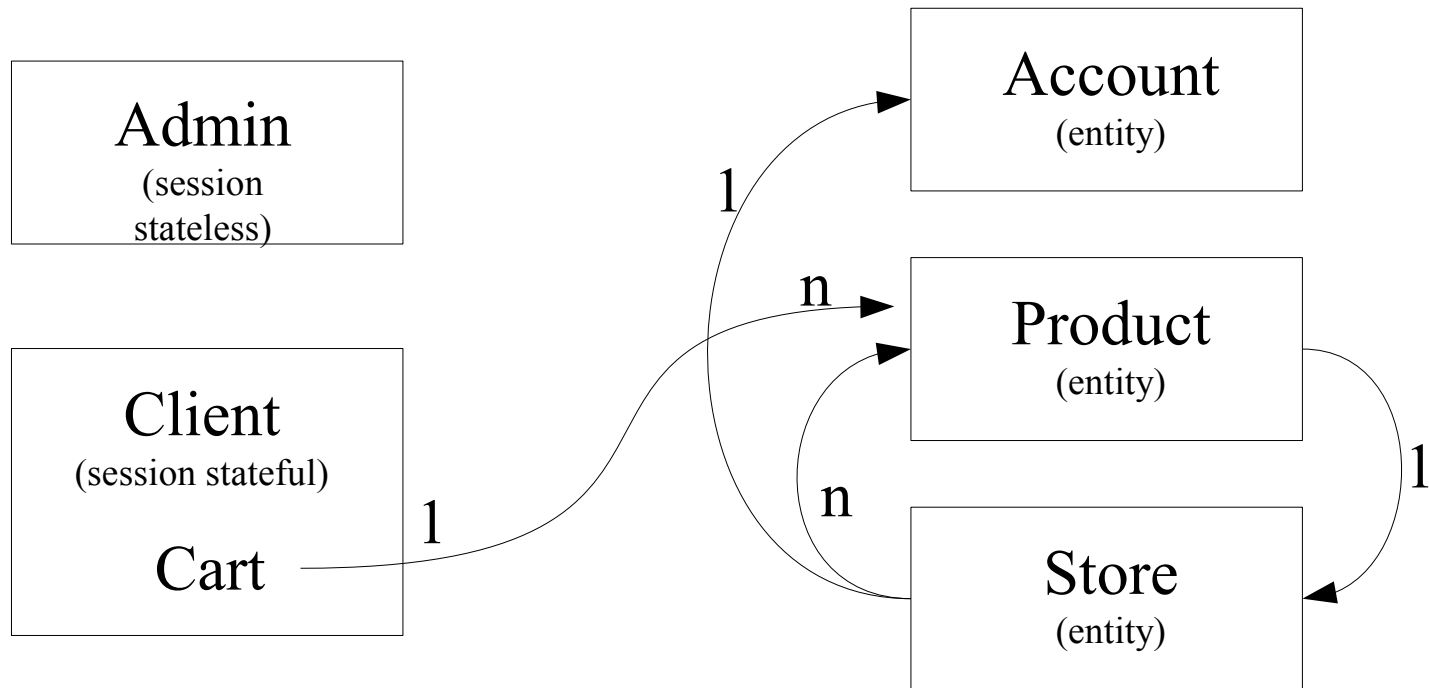
Objectifs du projet



- Aller un peu plus loin que le TP !
- Implanter une petite application de commerce électronique
- Découvrir par vous même des aspects en plus

L'application à implanter

- Une application de commerce électronique



Account



■ Champs

- int accountId (PK)
- String owner
- int balance

■ Interface

- int getAccountId();
- String getOwner();
- int getBalance();
- void deposit(int amount);
- int withdraw(int amount);

Product



■ Champs

- int productId (PK)
- String name
- int price
- Store store (ref de Store)

■ Interface

- int getProductId();
- String getName();
- int getPrice();
- Store getStore();

Store

■ Champs

- int storeId (PK)
- String name
- String cityName
- Account account (ref de Account)
- Collection products (ensemble de ref de Product)

■ Interface

- int getStoreId();
- String getName();
- int getCityName();
- Account getAccount();
- Collection getProducts();

Admin



- `int createAccount(String owner, int balance);`
- `int createStore(String name, String cityName, int accountId);`
- `int createProduct(String name, int price, int storeId);`

Client



- `Collection getStores();`
 - `storeId, name, city`
- `Collection getProductsFromStore(int storeId);`
 - `productId, name, price`
- `void addProduct(int productId); // in cart`
- `void removeProduct(int productId); // from cart`
- `Collection getProducts(); // in cart`
- `int getTotalPrice(); // of cart`
- `void buy(int accountId); // the cart`

Autonomie



- Des aspects n'ont pas été détaillés en cours/TP
 - Requêtes du gestionnaire de persistance (EJB-QL)
 - Transactions
 - ...
 - Documentez vous !
 - Tutorial Jboss
- Les interfaces données précédemment ne sont qu'un exemple
- Faites une belle application de démonstration, surtout claire et bien structurée
- Utiliser d'autres technologies : Glassfish, Struts, Ajax ...

Déroulement du projet



- En quadrinome (dans groupe TD)
- RDV obligatoire à chaque séance pour faire le point avec l'enseignant (moi)
- Evaluations : 7 juin
- Un rapport doit être rendu à la fin avec le code

Conseils



- Ne commencez pas la veille de la 1ère séance
- Vous êtes 4, il y a une raison, répartissez vous les tâches
- Démarrez avec des petits tests (tâches)
 - Faire un environnement propre de développement (jboss, glassfish, eclipse, netbean ...)
 - Faire marcher un exemple de bout en bout
- Travaillez (et debuggez) par incréments (et sauvegardes)

Prochaines étapes



- Constitution des quadrinomes (maintenant)
- 1ere réunion de suivi
 - Définition du site (nature, architecture)
 - Plan de travail et répartition des tâches
 - Se documenter
 - Lever certains verrous techniques (petits tests locaux)