

Nouveautés

Beans Session

Dans la nouvelle spécification, un EJB session ne doit définir qu'une interface métier et une classe d'implémentation, l'interface home ayant disparu.

Ces deux fichiers sont de simples objets java qui n'implémentent ou n'héritent d'aucun autre fichier particulier.

La configuration du descripteur de déploiement se trouve maintenant dans le code source java en utilisant les annotations prévues à cet effet. Il est cependant possible de continuer à utiliser le fichier de configuration pour les développeurs qui ne souhaitent pas les utiliser.

Bean Entité

Nombreux changements, notamment dans la manière d'accéder au composant: les recherches JNDI pour localiser un EJB sont remplacées par l'injection de dépendance.

MessageDriven beans

Dans la spécification 2.1, un message driven bean implémente la MessageDrivenBean interface et le message listener interface. Toutes les méthodes sont implémentées dans le bean lui même et il n'existe pas d'interface locale, Remote ou encore Home.

Dorénavant, la configuration du composant peut se faire par le biais des annotations mais peut toujours être contenue dans le descripteur de déploiement.

Avantages/inconvénients

Les EJB 3 sont plus faciles à développer que les 2.1. La simplification vient grandement de l'assemblage des EJB: `ejb-jar.xml`, interfaces Home, ...

Mais cela induit un fort couplage entre les métiers.

La personne qui assemble l'application en terme de transaction, de sécurité ou d'organisation des tables SQL doit dorénavant modifier les sources Java.

Si on veut éviter ce couplage, il faut repasser par les fichiers de descriptions xml externes (`ejb-jar.xml`).

Un autre inconvénient des EJB3 est leur disponibilité dans les conteneurs. Tous ne supportent pas encore les EJB3, et si ils les supportent, ne sont pas forcément compatible entre les versions.

Par exemple, en jboss4.2, il faut utiliser:

```
import org.jboss.annotation.ejb.*;
```

et en v5:

```
import org.jboss.ejb3.annotation.*;
```

et en v6:

```
import javax.ejb.*;
```