

## Mise en œuvre de serveurs d’application — TD n° 1

### 1 Introduction

Le but de ces TDs est de se familiariser avec les serveurs d’application. On se concentrera principalement sur les serveurs de type J2EE.

L’ensemble des TDs se fera sous environnement linux. On utilisera l’environnement de développement intégré eclipse, et le serveur d’application JBoss.

Dans ce premier TD vous découvrirez les serveurs d’application, à travers leur installation et leur utilisation, ainsi que l’environnement eclipse. Vous développerez ensuite votre première application J2EE.

### 2 Serveurs d’applications

#### 2.1 Installation

Dans la suite de ces TDs, on utilisera le serveur d’application JBoss.

1. Allez sur le site de JBoss : <http://www.jboss.org/>
2. Repérez comment télécharger une version. Nous cherchons un serveur d’application J2EE.

La version que nous allons utiliser est la 4.2.2.GA. Il n’est pas nécessaire de la télécharger, car le serveur est déjà installé localement. (Vous n’auriez de toute façon pas suffisamment de place sur votre compte pour le faire.)

3. Recherchez la documentation de JBoss. En particulier, trouvez (sans chercher à effectuer ces tâches) comment il est possible de :
  - installer le serveur d’application, après téléchargement ;
  - lancer le serveur ;
  - arrêter le serveur ;
  - lancer le serveur avec une configuration différente.
4. La version que nous utiliserons est installée dans le répertoire `/home_PC/bure1/jboss-4.2.2.GA/`. Utilisez soit les commandes Unix, soit le navigateur graphique pour parcourir l’arborescence. On retrouve les sous-répertoire suivants :
  - `bin` contient les scripts permettant entre autre de lancer le serveur ;
  - `client` contient les bibliothèques nécessaires pour créer des applications clientes (i.e. des applications qui utiliseront les services proposés par les applications sur le serveur) ;

- `docs` contient la documentation ;
- `lib` contient des bibliothèques utilisées par le serveur ;
- `server` contient les différentes configurations du serveur, avec leur contenu.

Quelles sont les trois configurations présentes ?

Dans une configuration donnée, les applications fournies seront mises dans le sous-répertoire `deploy`. Il s'agira de fichiers de type `.ear` (*Enterprise ARchive*, contient une application complète), `.war` (*Web ARchive*, uniquement la couche web), `.rar` (*Resource adaptater ARchive*, pour la connexion entre différentes ressources), ou encore `.sar` (*Service Archive*, utilisée par JBoss uniquement).

Pour installer une application, il suffit de la copier dans ce répertoire. Elle est mise en œuvre par le serveur « à chaud », c'est-à-dire qu'il n'y a pas besoin de le redémarrer pour qu'elle soit disponible. Nous regarderons le contenu de ces fichiers `.ear` dans des TDs ultérieurs.

5. Il existe beaucoup d'autres implantations du standard J2EE. Visitez les sites de quelques unes d'entre elles, et repérez les procédures de téléchargement et d'installation :
  - JOnAS : <http://jonas.objectweb.org/>
  - Geronimo : <http://geronimo.apache.org/>

## 2.2 Utilisation de JBoss

Une instance de la version installée de JBoss a normalement été lancée. Pour le vérifier,

1. Ouvrez un navigateur web (par exemple firefox).
2. Allez à l'adresse `http://localhost:8080/`

Si le serveur est bien lancé, une page intitulée « Welcome to JBoss™ » devrait s'afficher.

3. En cliquant sur JBoss Web Console, on peut obtenir des informations sur le serveur.

En particulier, on peut développer J2EE Domains (en cliquant sur le petit rond à gauche), puis `jboss.management.local`, puis JBoss (<http://www.jboss.org/>)... pour accéder à la liste des applications et services disponibles sur le serveur. En particulier, on voit dans la liste `web-console.war`, qui est justement l'application que nous sommes en train d'utiliser pour afficher ces pages. Elle est fournie avec d'autres par défaut lors de l'installation du serveur JBoss.

## 3 Découverte de l'environnement de développement intégré eclipse

Par la suite, nous utiliserons l'environnement de développement intégré (IDE) eclipse (<http://www.eclipse.org/>). Il s'agit d'une interface graphique qui permet de gagner du temps en automatisant certaines tâches et en fournissant des services utiles (complétion automatique par exemple). Nous utiliserons plus particulièrement la version qui contient des outils adaptés à J2EE (*Eclipse IDE for Java EE Developers* sur le site de téléchargement).

Ici aussi, eclipse est déjà installé sur le système. Toutefois il vous faut le configurer pour votre compte, en suivant les instructions suivantes :

1. Lancez eclipse en tapant `/home/prof/bure1/eclipse/eclipse` dans une console.

2. On vous demande de choisir un espace de travail. Laissez la valeur par défaut en cliquant sur **Ok**.
3. Cliquez sur la flèche à droite pour accéder à eclipse directement (**Workbench**).
4. Nous allons rajouter le serveur JBoss dans la configuration.
  - (a) Cliquez sur l'onglet **Servers** dans la partie en bas à droite.
  - (b) Cliquez droit dans la partie blanche en dessous. Faites **New ► Server**.
  - (c) Développez JBoss, sélectionnez JBoss v4.2. Cliquez sur **Next**.
  - (d) Entrez `/home_PC/bure1/jboss-4.2.2.GA` dans **Application Server Directory**.
  - (e) Cliquez sur **Finish**.

Un server JBOSS devrait apparaître, dans l'état **Stopped**. Pour le lancer, il faudrait cliquer dessus, puis sur la flèche blanche sur cercle vert au-dessus à droite. Néanmoins ceci ne marchera pas, car le serveur est déjà lancé en dehors d'eclipse. En effet, nous utilisons tous la même instance du serveur, pour des raisons pratiques. Pour mettre à jour les applications que vous mettrez sur le serveur, vous vous contenterez d'utiliser la cinquième icône (celle avec un serveur et des feuilles de papier, **Publish to the server**).

5. Il nous faut également dire à eclipse comment trouver un outil que nous utiliserons par la suite, nommé XDoclet (<http://xdoclet.sourceforge.net/>). Il est déjà installé dans le répertoire `/home/prof/bure1/xdoclet-1.2.3/`.
  - (a) Allez dans le menu **Window** puis **Preferences**.
  - (b) Cliquez sur **XDoclet**.
  - (c) Entrez `/home/prof/bure1/xdoclet-1.2.3/` dans **XDoclet Home**.
  - (d) Choisissez la version **1.2.3**.
  - (e) Cliquez sur **Ok**.

## 4 Votre première application

Nous allons créer une nouvelle application, qui se contentera uniquement d'afficher une page (statique).

### 4.1 Création

1. Allez dans le menu **File**, choisissez **New ► Other**.
2. Développez **J2EE**, choisissez **Entreprise Application Project**. Cliquez sur **Next**.
3. Dans **Project name**, entrez `TD1_votre_login`. **Attention!** Comme nous travaillons tous sur le même serveur, il est important que le nom de votre projet soit différent de celui de vos camarades, d'où l'idée d'utiliser soit votre nom de famille, soit encore mieux votre login. Cliquez sur **Next** deux fois.
4. Cliquez sur **New Module**.
5. Comme notre application se contentera d'afficher des pages web, seul un module web est nécessaire. Décochez **Application client module**, **EJB module** et **Connector module**. Cliquez ensuite sur **Ok**.
6. Cochez **Generate Deployment Descriptor**, puis cliquez sur **Finish**.

## 4.2 Ajout sur le serveur

L'application est maintenant créée. Nous allons l'ajouter sur le serveur.

1. Cliquez droit sur le serveur JBOSS. Choisissez Add and Remove Projects.
2. Cliquez sur votre projet, puis sur Add >.
3. Cliquez sur Finish.

On remarque que l'on peut maintenant développer le serveur JBoss (petit triangle à gauche). On retrouve alors votre projet, en italique car il n'est pas à jour sur le serveur.

4. Pour le publier sur le serveur, cliquer sur la cinquième icône au-dessus à droite (celle avec un serveur et des feuilles de papier, Publish to the server).
5. Si tout se passe bien, aucun message d'erreur ne devrait apparaître, et au bout d'un moment vous devriez être dans l'onglet Console. En revenant dans l'onglet Servers, on voit que l'application est maintenant synchronisée.

## 4.3 Ajout d'une page statique

Votre application est normalement disponible à l'adresse

`http://localhost:8080/TD1_votre_loginWeb/`

1. Utilisez firefox pour y accéder.  
Une page d'erreur 404 est affichée, vous informant que la ressource demandée n'est pas disponible. En effet, nous n'avons pas encore créé de page web dans notre application.
2. Retournez dans eclipse. Dans le Project Explorer à gauche, cliquez droit sur TD1\_votre\_loginWeb. Choisissez New ► HTML.
3. Dans File name entrez `index.html`. Cliquez sur Next.
4. Choisissez le template New XHTML File (1.0 strict). Cliquez sur Finish.
5. Remplacez `Insert title here` par `Ma première application`.
6. Après `<body>`, ajoutez `<p>Rien de particulier à dire.</p>`
7. Enregistrez (icône disquette).
8. Republiez (5<sup>e</sup> icône de l'onglet Servers).
9. Retournez dans firefox et visitez l'adresse `http://localhost:8080/TD1_votre_loginWeb/` à nouveau.

Votre page devrait maintenant apparaître.

Pour l'instant votre application ne fait pas grand chose, elle se contente d'afficher une page HTML qui n'est même pas dynamique. Pour cela, un simple serveur web, par exemple apache, aurait suffi. Nous verrons dans les prochains TDs comment faire des pages dynamiques en utilisant JSP, comment utiliser les services de la logique métier, etc.