

# Réalisation d'un composant Web Java EE

## Les packages WAR

Les composants web (servlets ou JSP) sont packagés pour le déploiement dans un fichier d'archive standard Java appelé war (Web Archive). Un fichier war a une structure arborescente spécifique. Le répertoire de plus haut niveau est le document racine de l'application.

Le document racine est celui contenant les pages JSP, les classes et archives de la partie client ainsi que les ressources web statiques. Il contient le sous-répertoire appelé WEB-INF qui contient lui-même les fichiers et répertoires suivants :

web.xml : dans le cas d'un composant servlet, le descripteur de déploiement standard XML au format défini par les spécifications Java EE.

classes : un répertoire contenant les servlets et les classes utilitaires.

## Démarche générale de réalisation de composant web

- 1 - Créer la structure de répertoires nécessaire
- 2 - Ecrire les pages, ou les classes en Java dans le cas d'un composant servlet, puis les compiler en plaçant les .class dans le répertoire adéquat.
- 3 - Ecrire le fichier xml de configuration (pour un composant servlet).
- 4 - Créer une archive.
- 5 - Déployer le composant sur le serveur d'application.
- 6 - Utiliser les composants via un navigateur web.

## Utilitaires

### 1 - Création d'une archive

```
jar cvf nom_archive.war Liste-fichiers-a-archiver
```

### 2 - Consultation du contenu d'une archive

```
jar tvf nom_archive.war
```

### 3 - Compilation dans un répertoire cible

```
javac -d repertoire_cible Liste-fichiers-a-compiler.java
```

Le répertoire cible peut être exprimé en absolu ou en relatif.

# Utilisation du serveur d'application Glassfish

## 1 - Configuration (éventuelle) de l'environnement Linux pour utiliser le serveur d'application

Ajouter au fichier .bashrc de votre répertoire de connexion les deux lignes suivantes :

```
export PATH=$PATH:/usr/local/stow/glassfish4/glassfish/bin/
export CLASSPATH=$CLASSPATH:./usr/local/stow/glassfish4/glassfish/modules/
gf-client-module.jar:/usr/local/stow/glassfish4/glassfish/lib/appserv-
rt.jar:/usr/local/stow/glassfish4/glassfish/lib/javaee.jar
```

## 2 - Création d'un domaine pour le serveur d'application (une seule fois !)

Dans votre environnement de travail, créer un répertoire nommé domains, puis en ligne de commande, créer le domaine en choisissant toutes les options par défaut :

```
asadmin create-domain --domainindir chemin_absolu_du_repertoire_domains domain1
```

## 3 - Démarrage de Glassfish dans votre domaine

```
asadmin start-domain --domainindir chemin_absolu_du_repertoire_domains
```

## 4 - Déploiement d'un composant web

Dans un navigateur, accéder à l'adresse : <http://localhost:4848/>

Le déploiement se fait à partir de sous-menu Application

## 5 - Utilisation d'un composant web :

[http://nomdemachine:numérodeport/nomdeContexte/page\\_JSP](http://nomdemachine:numérodeport/nomdeContexte/page_JSP) ou URL\_Servlet

Le nom de votre machine sera localhost et le numéro de port 8080.

Le nom du contexte représente le nom de l'archive, sans l'extension.

Le nom de servlet est celui qui a été précisé dans la balise url-pattern du fichier web.xml.

## Descripteur de déploiement web

Le descripteur de déploiement standard (fichier web.xml du répertoire WEB-INF) contient des informations décrivant les éléments à déployer :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web
  Application 2.3//EN" 'http://java.sun.com/dtd/web-app_2_3.dtd'>
<web-app>
</web-app>
```

Les servlets à déployer sont décrites par les balises suivantes :

```
<servlet>
  <servlet-name>NomInterneServlet</servlet-name>
  <servlet-class>NomClasseServlet</servlet- class >
</servlet>
<servlet-mapping>
  <servlet-name>NomInterneServlet</servlet-name>
  <url-pattern>/Accueil</url-pattern>
</servlet-mapping>
```