
Présentation personnelle

Jetty, Tomcat, Orion, JBoss et le reste...

Auteur : Lucas PRÊTRE

Classe : ETR6

Table des matières

1	Introduction	2
2	Bajie server	3
3	Jetty	5
4	JBoss	7
5	Resin	8
6	Apache Tomcat	9
7	Orion	10
8	Conclusion	11
	Références	12

1 Introduction

Il existe un nombre impressionnant de serveurs Web en “freeware”, parmi lesquels Apache Tomcat est le plus célèbre. Il est alors intéressant d’en savoir un peu plus sur ces fameux serveurs web et quelles sont leurs principales caractéristiques, au niveau des servlets et JSP’s. Ce document a pour objectif un certain aperçu des principaux serveurs web et non d’être exhaustif. Il est donc vivement conseillé, pour plus d’informations, de consulter les sites internet respectifs des différents serveurs web.

2 Bajie server

Description

*Bajie server*¹ est un moteur de JSP et servlets rapide ainsi qu'un serveur HTTP autonome. Il est le plus petit dans son genre, incluant tout de même une large palette de fonctionnalités. Il peut également servir comme *plugin* de moteur de JSP et servlets pour Apache ou IIS via AJP². Il est décrit comme suit :

- Gratuit.
- Rapide : 3 fois plus rapide que Tomcat (le moteur "standard" pour des JSP).
- Portable : il s'exécute sur toute machine où Java est installé (Windows, Linux, UNIX, Mac, etc...).
- Répandu : *Bajie server* compte des utilisateurs hors des Etats-Unis, les fonctions employant *Unicode* sont testées et supportées.

Bien que ces adjectifs soient élogieux, il reste encore à prouver la rapidité, qui peut devenir une notion subjective assez rapidement ! Néanmoins, les mesures effectuées démontrent effectivement que ce serveur web - pour les JSP's et servlets - est jusqu'à trois fois plus rapide que le célèbre Apache Tomcat.

Caractéristiques

Les principales caractéristiques du serveur sont les suivantes :

- Supporte les Servlets 2.3, JSP 1.2, War (Web Application ARchive - fichier contenant tous les composants pour une application web) et les tag libs.
- Connecteur AJP standard pour travailler avec Apache et IIS.
- Debugger simple intégré pour JSP / Servlet.
- Rechargement automatique des servlets, War et JSP's.
- Développement de servlet facile : copier la classe Java dans le répertoire *servlets* et c'est bon !
- Chargement de classes configurable : on définit le ou les répertoires de travail puis la recompilation des classes sera détectée par le serveur qui alors se rechargera.

¹Pour l'anecdote, Bajie est le nom d'un célèbre cochon Chinois qui apprécie les jolies filles et qui possède certains pouvoirs magiques. En fait, la raison invoquée est simple : l'auteur souhaitait que le serveur soit aussi simple que possible, pour des programmeurs paresseux ;-)

²AJP : protocole utilisé par un *web connector* pour communiquer avec un élément *AJP Connector*. On les utilise dans les cas où l'on désire intégrer, par exemple, Tomcat dans une installation existante d'Apache. En règle générale, cela est préférable pour des raisons de performance.

- Implémente les *Virtual Hosts*³.
- Gestion aisée du serveur grâce à une servlet de configuration.
- Le développement de servlets et de wars est facilité. Il suffit de glisser les servlets dans le chemin des classes du serveur (par défaut dans `htdocs/WEB-INF/classes` - il est possible de configurer le serveur pour avoir plusieurs répertoires pour différents groupes de servlets). Les fichiers `.war` peuvent être déposés dans le répertoire `wars/` et seront accessibles par une URL directe (exemple : on suppose que le fichier `.war` se nomme `testWar.war`, alors il sera accessible en tapant :
`http://localhost/testWar/`. `testWar` est appelé *War context name*. Par défaut, celui-ci est le nom du fichier, mais il est possible d'en définir d'autres par configuration.
- Cookies conformes à la RFC 2109 (HTTP State Management Mechanism)
- Implémente les scripts CGI
- Possibilité d'ajouter HTTPS en téléchargeant JSSE gratuitement depuis Sun.
- Implémente les *Virtual Path*.
- Inclue la personnalisation des pages d'erreurs avec une servlet.
- Implémente un cache.
- Inclue la notion de session lorsque les cookies ne sont pas acceptés ou supportés.

L'URL <http://www.geocities.com/gzhangx/websrv/index.html> donne accès au téléchargement du serveur, à une foire aux questions très précieuse, à un guide d'installation et à d'autres informations plus détaillées.

³Plusieurs noms de domaine (ou adresses IP) pour le même serveur web

3 Jetty

Description

Jetty est un serveur HTTP et un moteur de servlet écrit en Java. Il n'est donc pas indispensable de configurer et exécuter un serveur web séparé (comme Apache) afin d'utiliser les servlets et JSP pour générer des pages avec du contenu dynamique. Ainsi, le serveur web et le serveur d'application s'exécutent dans le même processus évitant donc des complications d'interconnexion. De plus, étant un composant Java pur, Jetty peut simplement être inclus dans une application pour des démonstrations, pour une distribution ou pour un déploiement. Jetty est disponible pour toutes les plateformes supportant Java.

Jetty est développé avec les conseils de *Mort Bay Consulting* et publié sous licence *Open Source*. Les sources sont incluses dans chaque version de Jetty. La licence impose quelques restrictions quant à son utilisation. Il est gratuit, que ce soit pour une utilisation commerciale ou pour une distribution.

Jetty a été optimisé en l'utilisant pour des applications commerciales et expérimentales depuis 1995 et le résultat est un petit serveur performant :

- Le serveur HTTP/1.1 peut être configuré dans un fichier Jar plus petit que 300Kb.
- Jetty est un des moteurs de servlets les plus rapides (selon les benchmarks).
- Il supporte plusieurs milliers de connexions simultanées sans problème.
- Les performances se dégradent avec "élégance" lors d'une montée en charge.

Jetty est largement utilisé dans différents projets et applications (commerciaux ou open source) allant des environnements PDA's (organiseurs personnels) à de très grandes unités centrales. Il peut être également embarqué dans des applications où HTTP est seulement un autre protocole d'interface sans adopter une architecture focalisée sur une application WWW.

Le site de Jetty <http://jetty.mortbay.com> contient d'autres informations plus complètes.

Caractéristiques

Les principales caractéristiques du serveur sont les suivantes :

- Supporte les Servlets 2.3, JSP 1.2, War (Web Application ARchive - fichier contenant tous les composants pour une application web) et les tag libs.
- Connecteur AJP standard pour travailler avec Apache et IIS.
- Implémente les *Virtual Hosts*
- Implémente les scripts CGI
- Petit compensateur de charge (*load balancing*) intégré.
- Supporte les fichiers Web Application Archives (fichiers WAR).
- Possibilité d'ajouter HTTPS en téléchargeant JSSE gratuitement depuis Sun.

4 JBoss

JBoss est serveur d'application *Open Source*. Gratuit, il est écrit en Java et inclue toutes les fonctionnalités de base J2EE. Il peut être utilisé à partir d'environnements monoprocesseurs Linux/Windows jusqu'à des environnements contenant des centaines de processeurs comme des clusters Solaris. Cette flexibilité fait partie intégrante du domaine de développement dans lequel JBoss Group est expert.

JBoss évolue de manière permanente. En effet, une fois par mois environ une nouvelle version de développement est publiée et tous les six mois en moyenne, une version de production est mise à disposition. Il est intéressant de souligner que JBoss a gagné le prix du "best java app-server" dans les "JavaWorld Editors' Choice" awards en 2002. JBoss est utilisé tout autour du globe dans des entreprises et gouvernements.

Comme on peut le voir sur la figure 1, JBoss existe en deux variantes soit en intégrant le serveur HTTP Jetty ou le serveur HTTP Apache Tomcat. Les caractéristiques du serveur web sont donc celles de Jetty ou celles de Tomcat. Pour plus de détails concernant JBoss, il est possible d'aller consulter son site internet : <http://www.jboss.org>

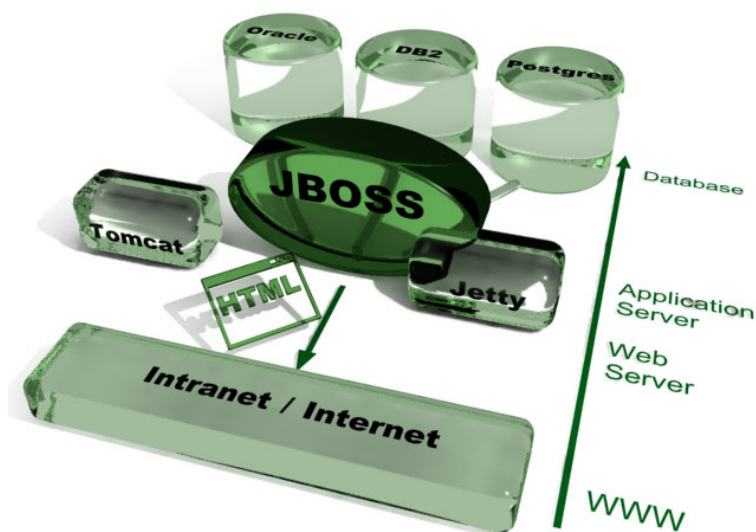


FIG. 1 – Architecture JBoss

5 Resin

Description

Resin est un moteur rapide de servlets et JSP qui supporte la compensation de charge (*load balancing*) pour une fiabilité accrue. La philosophie de Resin est de séparer le contenu de la mise en forme. Ainsi, les servlets peuvent générer de l'XML simple et utiliser ensuite un filtre de mise en forme XSL pour formater le résultat pour correspondre aux capacités de chaque client, des palm pilots à Mozilla. Il inclue un serveur web HTTP/1.1 complet dédié pour fournir rapidement du contenu Java dynamique. Bien que Resin soit amélioré pour du contenu dynamique, ses performances pour du contenu statique sont égales à celles d'Apache. La plupart des sites utilisent le serveur web Resin pour toutes leurs requêtes HTTP.

Resin implémente les *virtual hosts*. Certains sites ont besoin de fonctionnalités spéciales d'autres serveurs web ou dont la politique d'entreprise requiert un autre serveur et ne peuvent donc pas profiter du serveur web intégré dans Resin. C'est pourquoi Resin fonctionne avec la plupart des serveurs web existants comme Apache, IIS, iPlanet, Zeus et tout autre serveur web fournissant les interfaces NSAPI ou ISAPI. Le *virtual hosting* de Resin est supporté autant bien dans le cas de son propre serveur web ou en conjonction avec d'autres serveurs web.

Resin est gratuit uniquement pour les étudiants, sinon il est payant et coût \$500 par serveur. Des information complémentaires et plus détaillées sont disponibles sur le site de Resin :
<http://www.caucho.com/>

Caractéristiques

Les principales caractéristiques du serveur sont les suivantes :

- Supporte les Servlets 2.3 et les JSP 1.2 .
- Simplification de la création de classes Java avec compilation et rechargement automatiques des codes sources Java lorsque ces derniers changent. Ainsi, la programmation de *JSP beans* implique seulement l'édition de code Java et l'appui sur le bouton "rafraîchir" du navigateur.
- La compensation de charge augmente la performance et la fiabilité.

6 Apache Tomcat

Description

Tomcat est un moteur de servlet et de JSP's. Il implémente les spécifications standards des servlets et JSP, mais également d'autres fonctions additionnelles qui font de lui une plateforme pour le développement et le déploiement de services et applications web. Tomcat fait partie d'un projet Apache - Jakarta. *The Apache Software Foundation* fournit de l'aide à la communauté Apache pour des projets de logiciels open source. Les projets Apache sont caractérisés par un processus de développement collectif, une licence open-source et le désir de créer des logiciels de haute qualité dans le domaine concerné. Toute personne intéressée est la bienvenue dans *The Apache Software Foundation*.

Tomcat utilise donc le serveur HTTP Apache et hérite donc de ses propriétés. Actuellement, deux versions sont les plus répandues : la version 1.3 et la version 2.0 ; voici les différences apportées par cette nouvelle version :

- Support de *Unix Threading*.
- Support d'IPv6.
- Configuration simplifiée.
- Nouvelle API Apache.
- Amélioration des modules.

Apache Tomcat est moins performant qu'Apache pour délivrer des pages statiques. Il est également moins configurable et moins robuste qu'Apache tout seul. C'est pourquoi il est recommandé d'utiliser Apache avec Tomcat comme moteur séparé de servlet et JSP. Pour plus d'informations au sujet d'Apache Tomcat, consulter son site à l'URL suivante :
<http://jakarta.apache.org/tomcat/>

Caractéristiques

Les principales caractéristiques du serveur sont les suivantes :

- Supporte les Servlets 2.3, JSP 1.2 et les tag libs.
- Connecteur AJP standard pour travailler avec Apache et IIS.
- Implémente les scripts CGI
- Possibilité d'ajouter HTTPS en téléchargeant JSSE gratuitement depuis Sun.

7 Orion

Description

Le serveur d'application Orion a été créé par *Ironflare AB*, entreprise se situant en Suède dont il est le seul produit.

Orion s'exécute sur la base de Java 2 et donc toute plateforme comportant une machine virtuelle Java peut exécuter Orion. Celui-ci a été testé sur Windows, Sparc Solaris, Linux, FreeBSD, AS/400 et HP-UX.

Plusieurs critères font d'Orion un des leader parmi les serveurs d'application, dont :

- Performances excellentes.
- Facilement un des serveurs d'application J2EE les plus rapides.
- Premier serveur disponible sur le marché avec toutes les fonctionnalités J2EE supportées.
- Caractéristiques uniques de développement : *auto-deploy* et *hot-swap*
- Prix attractif.

Les sources d'Orion ne seront pas portées vers une licence de type Linux dans un proche avenir. En effet, si *Ironflare AB* le faisait, elle serait poursuivie par Sun car la spécification de J2EE leur appartient et ne peut être divulguée.

Orion est gratuit pour toute utilisation non-commerciale. En revanche, pour une utilisation commerciale, le prix se monte à \$1500 par serveur déployant Orion (le nombre de CPU's n'entre pas en ligne de compte). Pour plus de détail sur le serveur Orion, consulter le site internet à l'URL suivante : <http://www.orionserver.com>

Mise à part le support des Servlets 2.3 et des JSP 1.2, les caractéristiques d'Orion ne sont pas clairement explicitées. En outre, il est possible d'utiliser un autre serveur HTTP et utiliser Orion comme moteur de servlets et de JSP's.

8 Conclusion

Les serveurs web se multiplient et il en existe une large palette avec pour chacun ses spécificités propres. Il est à noter que dans le cadre des servlets et JSP's, on remarque souvent que les serveurs traités dans ce document peuvent être utilisés comme serveurs web autonomes. Certains se lancent en moins de 10 secondes et ne nécessitent aucune installation car ils sont basés sur Java.

Pour la petite histoire, le serveur Bajie m'a convaincu car il se lance avec un fichier .bat (sans installation) et offre une petite interface graphique pour une configuration sommaire ainsi qu'une servlet de configuration assez complète. D'autre part, le site web de ce serveur web montre ses possibilités. Les données y sont donc facilement accessibles, ce qui n'est pas le cas d'autres serveurs, où les informations sont presque dissimulées. C'est pourquoi Bajie serveur est un peu mon "coup de coeur". Il n'a pas de renommée et doit donc montrer ce "qu'il a dans le ventre".

Il y a donc un reproche à faire à d'autres sites de serveurs web qui ne prennent plus la peine de mettre à disposition clairement les caractéristiques de leurs produits. C'est dommage, vraiment dommage. En conclusion, beaucoup de possibilités s'offrent pour exécuter des servlets et des JSP. Chaque cas d'utilisation mérite alors réflexion quant au choix du serveur web à utiliser.

Références

- [1] Bajie Http Web Server,
<http://www.geocities.com/gzhangx/websrv/index.html>
- [2] Jetty :// Web Server & Servlet Container,
<http://jetty.mortbay.com>
- [3] JBoss : : Professional Open Source
<http://www.jboss.org>
- [4] Caucho Technology - Resin
<http://www.caucho.com>
- [5] The Apache Jakarta Project - Tomcat
<http://jakarta.apache.org/tomcat/>
- [6] Orion Application Server website
<http://www.orionserver.com/>