**JBoss Enterprise Application Platform 6.4**

C:\Users\ADMIN\Downloads\1\Administration and Configuration Guide_files\title_logo.png

# JBOSS STANDALONE SERVER CONFIGURATIONS

## port offset

**Port offset** is an useful tweak which can be applied to execute several application servers on the same machine. A typical usage of the port-offset is for creating a **vertical cluster**, with multiple nodes on the same machine

**Standalone Server**

In "***standalone***" mode each JBoss Application Server 6.4 instance is an independent process. (It will be Single Standalone JVMs)

**Jboss Profiles:**

* + **Default** (No JMS,NO Clustering)- Java Enterprise Edition certified web profile configuration which includes the technologies required by the Web Profile specification plus Java Connector 1.6 Architecture, Java XML API for RESTFul Web Services, and OSGi (No JMS,No Clustering)- Used By Developer
  + **Full** (No Clustering)- referred to as our Java Enterprise Edition 6 full profile preview, this configuration contains all the technologies in the default configuration with additional technology previews of Java API for XML-Based Web Services (JAX-WS) 2.2 and Java Message Service 1.1. (No Clustering)
  + **Ha-High Availability** (No Jms)- default profile with Clustering capabilities(NO JMS)
  + **Full-ha**( With JMS,with Clustering)- EE 6 full profile preview with Clustering capabilities

##### **Standalone Server Configurations**

* **standalone.xml (default)** 
  + Java Enterprise Edition certified web profile configuration which includes the technologies required by the Web Profile specification plus Java Connector 1.6 Architecture, Java XML API for RESTFul Web Services, and OSGi (No JMS,No Clustering)
* **standalone-full.xml** 
  + referred to as our Java Enterprise Edition 6 full profile preview, this configuration contains all the technologies in the default configuration with additional technology previews of Java API for XML-Based Web Services (JAX-WS) 2.2 and Java Message Service 1.1. (No Clustering)
* **standalone-ha.xml** 
  + default profile with Clustering capabilities(NO JMS)
* **standalone-full-ha.xml** 
  + EE 6 full profile preview with Clustering capabilities
* **standalone-osgi.xml** 
  + EE 6 profile with osgi development capabilities

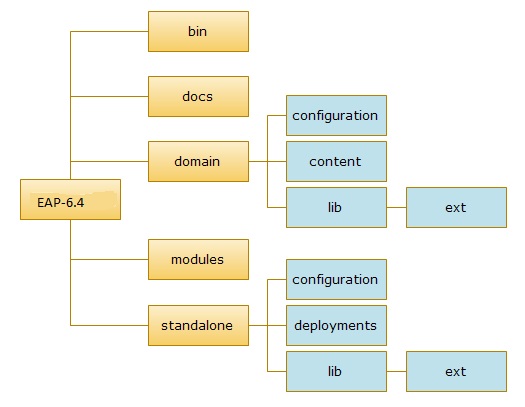
**Standalone Directory Structure**

In "***standalone***" mode each JBoss Application Server 6.4 instance is an independent process (similar to previous JBoss AS versions; e.g., 3, 4, 5, or 7). The configuration files, deployment content and writable areas used by the single standalone server run from a JBoss Application Server installation are found in the following subdirectories under the top level "standalone" directory:

|  |  |
| --- | --- |
| **DIRECTORY** | **DESCRIPTION** |
| configuration | Configuration files for the standalone server that runs off of this installation. All configuration information for the running server is located here and is the single place for configuration modifications for the standalone server. |
| data | Persistent information written by the server to survive a restart of the server |
| deployments | End user deployment content can be placed in this directory for automatic detection and deployment of that content into the server's runtime.  NOTE: The server's management API is recommended for installing deployment content. File system based deployment scanning capabilities remain for developer convenience. |
| lib/ext | Location for installed library jars referenced by applications using the Extension-List mechanism |
| log | standalone server log files |
| tmp | location for temporary files written by the server |

**The new server structure**

The first thing you'll notice when you browse through the application server folders is that its file system is basically divided into two core parts: the dichotomy reflects the distinction between standalone **servers** and **domain** servers.

****

**Where:**  
**-c** = is for server configuration file to be used  
**-b** = is for binding address  
**-u** = is for multicast address  
**-Djboss.server.base.dir** = is for the path from where node is present  
**-Djboss.node.name =** is for the name of the node  
**-Djboss.socket.binding.port-offset =** is for the port offset on which node would be running

**-Djboss.bind.address.management**= is for binding address

## port offset

**Port offset** is an useful tweak which can be applied to execute several application servers on the same machine. A typical usage of the port-offset is for creating a **vertical cluster**, with multiple nodes on the same machine