

CONTENTS INCLUDE:

- What is JBoss EAP 5?
- Installing EAP 5
- Directory Structure
- Administration
- Management
- Application Deployment and more...

Getting Started with JBoss Enterprise Application Platform 5

By Scott Marlow, Jaikiran Pai, Shelly McGowan, Brian Stansberry, and Len DiMaggio

WHAT IS JBOSS EAP 5?

JBoss Enterprise Application Platform is an open source implementation of the Java EE suite of services. It comprises a set of offerings for enterprise customers who are looking for preconfigured profiles of JBoss Enterprise Middleware components that have been tested and certified together to provide an integrated experience. Its easy-to-use server architecture and high flexibility makes JBoss the ideal choice for users just starting out with Java EE, as well as senior architects looking for a customizable middleware platform.

Because it is Java-based, JBoss Enterprise Application Platform is cross platform, easy to install and use on any operating system that supports Java. The readily available source code is a powerful learning tool to debug the server and understand it. It also gives you the flexibility to create customized versions for your personal or business use.

Visit the <http://www.jboss.com/products/community-enterprise/> website to download JBoss Enterprise Application Platform 5.

INSTALLING JBOSS EAP 5

1. Installation using the Graphical Installer

The graphical installer will guide you through the installation steps. Invoke the installer using the command:

```
java -jar enterprise-installer-5.0.1.GA.jar
```

2. Installation using the Zip Distribution

Use the unzip command:

```
unzip jboss-eap-5.0.1.GA.zip
```

Use the same steps to install optional native package:

```
unzip jboss-eap-native-5.0.1.GA-<operating-system>-<arch>.zip
```

DIRECTORY STRUCTURE

Contents of jboss-as:

Directory	Description
bin	Contains startup, shutdown and other system-specific scripts. All entry point JAR files and start scripts are here.
client	JAR files used by external Java client applications. Choose JAR files as required or use jbossall-client.jar
common/lib	Shared JAR files common to profiles are here.

docs	XML DTD and schemas for reference. Also example configuration files for setting up datasources (e.g. MySQL, Oracle, PostgreSQL).
lib	Contains server startup JARs and not intended to hold application JAR files.
server	Contains the JBoss server profile sets.

DIRECTORY STRUCTURE

JBoss EAP 5 configuration profiles are located within the jboss-as/server directory (specified with "run -c PROFILE"):

default	JavaEE 5 server profile. Has most frequently used EE services. Default does not include JAXR, IIOP, clustering services.
all	Bundles all services (including clustering and RMI/IIOP).
production	based on "all" profile, tuned for production; with log verbosity reduced, deployment scanning every 60 seconds, and memory usage tuned to accommodate production deployment requirements, configured to require authorization checks.
standard	based on 'all' profile and is a fully certified Java EE 5 configuration.
web	lightweight configuration created around JBoss Web and provides services required for web application deployment and a subset of Java EE technologies. Does not include JBoss Transaction JTS or XTS, Enterprise Java Bean 1.x or 2.x capabilities, JBoss Messaging, JCA, or JBoss IIOP
minimal	Bare minimum required to start. Includes logging, JNDI and URL deployment scanner. Use JMX/JBoss to start your own services. No web container, EJB or JMS support is included.

ADMINISTRATION

To start the JBoss EAP server, simply change to the EAP_DIST/jboss-as/bin directory. Set the environment variable JAVA_HOME. Execute the run.bat (for Windows) or run.sh (for Linux,



Need to integrate Spring, Hibernate, or Seam with JBoss AS?

We've already done it for you. JBoss Enterprise Application Platform 5.0 includes pre-integrated frameworks for building all sorts of Java apps.

- Customize your app server footprint
- Simplify your configurations
- Bring the power of full text search to your app with Hibernate Search
- Use our pre-integrated Apache CXF web services stack
- Seam, Hibernate, and Hibernate Search tooling through JBoss Developer Studio Hibernate queries

Download today: jboss.com/download

Unix, Mac OSX) script, as appropriate for your operating system.

Administering your JBoss EAP 5 server instance is easy with the administration consoles provided in this distribution. Once the server is started, simply point your browser to:

<http://localhost:8080>

This brings you to the consoles to manage your instance as well as links to on-line resource references.

Use the Administration Console for managing and monitoring a server instance. Deploy, undeploy, and update enterprise applications, persist configuration changes for Datasources, JMS topics and queues, Connection Factories, and Service Binding Manager, monitor standard JVM metrics, and view statistics and invoke operations on many other components.

The JMX Console provides a server view. It lists all registered services (MBeans). Administration Console shares the same username/password as JMX console.

Command-line scripts are available in the `jboss-as/bin` directory. In addition to scripts for starting and stopping the server, JBoss provides a command line tool that allows for interaction with a remote server instance. This tool is called **twiddle** (for twiddling bits via JMX) and is located in the `bin` directory. Twiddle is a command execution tool, not a general command shell. Run using the `twiddle.sh/twiddle.bat` scripts, and passing in a `-h(--help)` argument provides the basic syntax, and `--help-commands` shows commands. Twiddle defaults to the localhost at port 1099 to lookup the default `jmx/rmi/RMIAdaptor` binding of the `RMIAdaptor` service as the connector for communicating with the JMX server. To connect to a different server/port combination, can use the `-s (--server)` option:

```
$ ./twiddle.sh -s servername serverinfo -d jboss
$ ./twiddle.sh -s servername:1099 serverinfo -d jboss
```

MANAGEMENT

The JBoss Operations Network (JON) management platform delivers centralized systems management that allows you to:

- Coordinate stages of application's life-cycle datasources and messaging services
- Expose cohesive view of middleware components through complex environments
- Improve operational efficiency/reliability through visibility into production availability and performance
- Configure and roll out applications across complex environments through a single tool

APPLICATION DEPLOYMENT

There are multiple ways to deploy applications to JBoss:

Simplest way

- Choose the server profile to which you want to deploy the application (let's consider the "default" server profile in this example)
- Copy your application (for example: `.war` or `.ear` or a `.jar` file) to `JBOSS_HOME/server/<profilename>/deploy` folder.

- Start the server:

```
./run.sh
```

- JBoss will deploy your application when it boots up.
- That's it!

Hot Tip

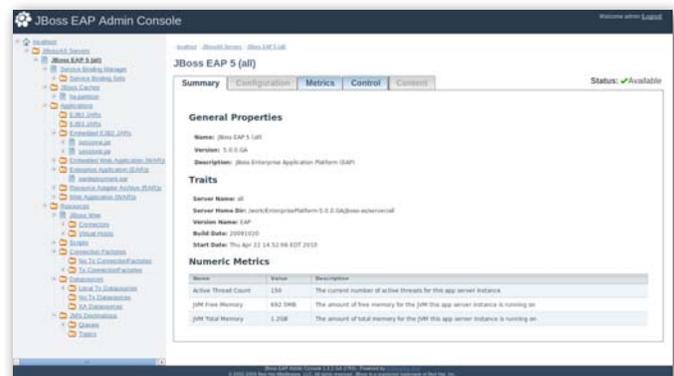
This approach does not require the server to be started when you are deploying your application. If you want to undeploy the application then just move (or delete) the application from the deploy folder. You can develop simple scripts [like an Ant script] to deploy the application to JBoss. All it takes is a file copy command.

Using the admin console

- Start the server

```
./run.sh
```

- Login to the admin console (<http://localhost:8080/admin-console>)



- On the left hand side of the admin console page, under JBoss AS -> Applications, select the type of application you want to deploy. Let's consider a web application (`.war`) in this example
- Click the "Web Application (WAR)" link
- In the right side section, under the "Summary" tab, click on "Add a new resource" button



- "Browse" to the `.war` file to deploy (e.g. `/home/me/myapp.war`)
- If you want to deploy the application in exploded format (instead of an archive) then select "Yes" radio button for "Deploy Exploded" option

- Click on the “Continue” button
- On successful deployment, you will see your application listed in the Summary tab
- To undeploy the application, click on the “Delete” button next to the application you want to undeploy, in the “Summary” tab

Hot deployment

JBoss has a built-in hot deployer which can:

- Detect new applications in the deploy folder and trigger an application deployment
- Detect an application which was removed from the deploy folder and trigger an application undeployment
- Detect that the deployment descriptor of an application (for example, the web.xml of .war or application.xml of .ear) has changed and trigger an application redeployment

Hot Tip

The hot deployer is configured to run every X milliseconds. This value can be changed by changing the “scanPeriod” attribute in `JBOSS_HOME/server/<profilename>/deploy/hdscanner-jboss-beans.xml`:

```
<!-- Hotdeployment of applications -->
<bean name="HDScanner" class="org.jboss.system.server.profileservice
.hotdeploy.HDScanner">
<property name="deployer"><inject bean="ProfileServiceDeployer"/></property>
<property name="profileService"><inject bean="ProfileService"/></property>
<property name="scanPeriod">5000</property>
<property name="scanThreadName">HDScanner</property>
</bean>
```

To disable hot deployment, remove the `hdscanner-jboss-beans.xml` from the deploy folder or rename it to `hdscanner-jboss-beans.xml.bak` (.bak files are ignored).

Clustering

Getting started with JBoss clustering is very simple. If two JBoss server instances using the all configuration are started on the same network, those servers will detect each other and automatically form a cluster.

Initial Preparation

Preparing a set of servers to act as a JBoss cluster involves a few simple steps:

- **Install JBoss on all your servers.**
- **For each node, determine the address to bind sockets to.**
- **Ensure multicast is working.** Make sure each server's networking configuration supports multicast and that multicast support is enabled for any switches or routers between your servers. JBoss clustering also offers non-default configuration options that do not use multicast.
- **Determine a unique integer “ServerPeerID” for each node.** JBoss Messaging requires that each node in a cluster has a unique integer id, known as a “ServerPeerID”, that should remain consistent across server restarts. A simple 1, 2, 3, ..., x naming scheme is fine.

Launching Your Cluster

We'll look at two scenarios for doing this. In each scenario we'll be creating a two node cluster, where the `ServerPeerID` for the first node is 1 and for the second node is 2.

Scenario 1: Nodes on Separate Machines

On node1, to launch JBoss:

```
$ ./run.sh -c all -b 192.168.0.101 -Djboss.messaging.ServerPeerID=1
```

On node2, it's the same except for a different -b value and `ServerPeerID`:

```
$ ./run.sh -c all -b 192.168.0.102 -Djboss.messaging.ServerPeerID=2
```

The -c switch says to use the all config, which includes clustering support. The -b switch sets the address on which sockets will be bound. The -D switch sets the system property from which JBoss Messaging gets its unique id.

Scenario 2: Two Nodes on a Single, Non-Multihomed, Machine

Running multiple nodes on a single machine that only has a single IP address is a common scenario in a development environment. You need to be sure each server instance has its own work area. One way to do this is to simply make copies of the all configuration. For example, assuming the root of the JBoss distribution was unzipped to `/var/jboss`, you could:

```
$ cd /var/jboss/server
$ cp -r all node1
$ cp -r all node2
```

Two processes can't bind sockets to the same address and port, so we'll have to tell JBoss to use different ports for the two instances. This can be done by setting the `jboss.service.binding.set` system property.

To launch the first instance, open a console window and:

```
$ ./run.sh -c node1 -b 192.168.0.101 -Djboss.messaging.ServerPeerID=1 \
-Djboss.service.binding.set=ports-default
```

For the second instance, in a second console window:

```
$ ./run.sh -c node2 -b 192.168.0.101 -Djboss.messaging.ServerPeerID=2 \
-Djboss.service.binding.set=ports-01
```

This tells the `ServiceBindingManager` on the first node to use the standard set of ports (e.g. JNDI on 1099). The second node uses the “ports-01” binding set, with which by default each port has an offset of 100 from the standard port number (e.g. JNDI on 1199).

Web Application Clustering Quick Start

Web application clustering involves two aspects: setting up an HTTP load balancer and telling JBoss to make the application's user sessions HA. How to do the former depends on what load balancer you choose (`mod_cluster` is a good choice); the latter couldn't be simpler – just add the `<distributed/>` to your application's `web.xml`.

EJB3 Session Bean Clustering Quick Start

To add load balancing and failover capabilities to your EJB3 session beans, simply add the `org.jboss.ejb3.annotation`. Clustered annotation to the bean class for your stateful or stateless bean:

```
@javax.ejb.Stateful
@org.jboss.ejb3.annotation.Clustered
public class MyBean implements MySessionInt {

    public void test() {
        // Do something cool
    }
}
```

PERFORMANCE AND TUNING

Identify peak application workload and difference from normal workload. In understanding peak workloads, don't go by averages as the peaks may be much more than the averages calculated over a period. Start testing with a low load and increase until expected peak load. Tune until target performance is achieved. There are a number of possible performance optimizations. Always load test before and after making changes to verify the intended effect. Make one change at a time so it's clear what change has what effect.

Use OS monitoring tools to identify system performance bottlenecks. In a multiple machine installation, find the machine(s) that are the bottleneck.

Instrument the application for performance measurement (make optional for production). Also, turn on container call statistics and Hibernate statistics.

Taking successive thread dumps indicates what is going on. Do this prior/after hitting a performance wall. Generate a thread dump once a minute over a five minute performance problem and compare your findings. Use "jps -l" command to get the Java process ids and then run the "jstack ProcessID" command (generates the thread dump.)

The HotSpot Java Virtual Machine contains other information gathering tools which can help tune applications. More information is in <http://java.sun.com/javase/technologies/hotspot/>.

jmap generates a memory heap dump file that can easily be read by the Eclipse Memory Analyzer tool (<http://www.eclipse.org/mat/>). **jstat** shows details of the memory space.

Clustering Tuning

Ensuring Adequate Network Buffers

Inadequately sized network buffers can cause lost packets. Steps to increase max buffer sizes are OS specific. For Linux change /etc/sysctl.conf file:

```
# Allow a 25MB UDP receive buffer for JGroups
net.core.rmem_max = 26214400
# Allow a 1MB UDP send buffer for JGroups
net.core.wmem_max = 1048576
```

Isolating Intra-Cluster Traffic

isolate intra-cluster traffic from external request traffic. This requires multiple NICs on your server machines, with request traffic coming in on one NIC and intra-cluster traffic using another.

```
./run.sh -c all -b 10.0.0.104 -Djgroups.bind_addr=192.168.100.104
```

JGroups Message Bundling

Message bundling queues small messages until a configurable number of bytes have accumulated, then sent as a large message. Use of bundling can have significant performance benefits for high-volume asynchronous session replication. However, it is not enabled by default, as bundling can add significant latency to other types of intra-cluster traffic, particularly clustered Hibernate/JPA Second Level Cache traffic.

To use a JGroups channel with message bundling enabled, edit the <JBoss_Home>/server/<profilename>/deploy/cluster/jboss-cache-manager.sar/META-INF/jboss-cache-jboss-beans.xml file. For example, for the cache used by default for web sessions:

```
...
<!-- Standard cache used for web sessions -->
<entry><key>standard-session-cache</key>
<value>
  <bean name="StandardSessionCacheConfig" class="org.jboss.cache.config.
  Configuration">
  ...
  <!-- Replace standard 'udp' JGroups stack with one that uses message
  bundling -->
  <property name="multiplexerStack">udp-async</property>
  ...
  ...
```

For FIELD granularity web sessions, in the same file the same change can be made to the cache configuration with the field-granularity-session-cache key. For EJB3 stateful session beans, in the same file the same change can be made to the cache configuration with the sfsb-cache key.

Enabling Buddy Replication for Session Caches

In a cluster of more than two nodes, you can improve performance by enabling "buddy replication" in the web session and stateful session bean caches. With buddy replication, instead of replicating a copy of sessions to all nodes in the cluster, a copy is only replicated to a configurable number of "buddy" nodes.

Buddy replication is enabled by editing the <JBoss_Home>/server/<profilename>/deploy/cluster/jboss-cache-manager.sar/META-INF/jboss-cache-jboss-beans.xml file. For example, for the cache used by default for web sessions:

```
...
<!-- Standard cache used for web sessions -->
<entry><key>standard-session-cache</key>
<value>
  <bean name="StandardSessionCacheConfig" class="org.jboss.cache.config.
  Configuration">
  ...
  <property name="buddyReplicationConfig">
    <bean class="org.jboss.cache.config.BuddyReplicationConfig">
    <!-- Just set to true to turn on buddy replication -->
    <property name="enabled">true</property>
  ...
  ...
```

For FIELD granularity web sessions, in the same file the same change can be made to the cache configuration with the field-granularity-session-cache key. For EJB3 stateful session beans, in the same file the same change can be made to the cache configuration with the sfsb-cache key.

Reducing the Volume of Web Session Replication

Reducing the amount data being replicated can obviously improve performance. This can be accomplished both by avoiding replication when a request hasn't actually updated the session and by limiting replication to only the session data that has actually changed. See the discussion of replication-trigger and replication-granularity in "http://www.redhat.com/docs/en-US/JBoss_Enterprise_Application_Platform/5.0.0/html/Administration_And_Configuration_Guide/clustering-http-state.html" for how to configure your application to limit the amount of data replicated.

Monitoring JGroups via JMX

When clustering services create a JGroups Channel to use for

intra-cluster communication, the JMX console will include the following:

```
jboss.jgroups:cluster=<cluster_name>,protocol=UDP,type=protocol
Statistics on two thread pools used to carry incoming messages
up the channel's protocol stack.
```

```
jboss.jgroups:cluster=<cluster_name>,protocol=UNICAST,type=protocol
Information on lossless, ordered delivery of unicast
(i.e. point-to-point) messages.
```

```
jboss.jgroups:cluster=<cluster_name>,protocol=NAKACK,type=protocol
Information on lossless, ordered delivery of multicast
(i.e. point-to-multipoint) messages.
```

```
jboss.jgroups:cluster=<cluster_name>,protocol=FC,type=protocol
Information on ensuring fast message senders do not
overwhelm slow receivers.
```

Web deployer

Other key configurations required for performance tuning of your Enterprise Application Platform include the `<JBoss_Home>/server/<your_configuration>/deployers/jbossweb.deployer/server.xml` file that sets your HTTP requests pool.

Thread pool

JBoss Enterprise Application Platform 5 has a robust thread pooling, that should be sized appropriately. The server has a `jboss-service.xml` file in the `<JBoss_Home>/server/<your_configuration>/conf` directory that defines the system thread pool. There is a setting that defines the behavior if there isn't a thread available in the pool for execution. The default is to allow the calling thread to execute the task. You can monitor the queue depth of the system thread pool through the JMX Console, and determine from that if you need to make the pool larger.

TROUBLESHOOTING

Startup problems

If you are having trouble starting JBoss, the first thing to check is the `JAVA_HOME` environment variable. This should point to the home of your JDK (or JRE installation). For example, if your JDK is installed at `/opt/Java/jdk1.5.0` then `JAVA_HOME` should be set as follows:

```
JAVA_HOME=/opt/Java/jdk1.5.0
```

On Windows OS, if your JDK installation is at `C:/Java/jdk1.5.0`, you can set it as follows:

```
set JAVA_HOME=C:/Java/jdk1.5.0
```



It's highly recommended not to install JBoss or Java in a folder containing a space in its path. For example, do not install Java at `C:/Program Files/Java/jdk1.5.0` on Windows OS.

Logs

JBoss by default is configured to log messages to the `JBOSS_HOME/server/<servername>/log.server.log` file. This

file can be checked for any exceptions or other informational logging. Logging levels can be controlled in `JBOSS_HOME/server/<servername>/conf/jboss-log4j.xml`

Thread dumps

Sometimes, if you notice that the application is not responding, you can generate thread dumps to check what each thread is currently doing. Thread dumps can be generated in multiple ways – 2 of which are explained below:

From jmx-console

- Access `jmx-console` (<http://localhost:8080/jmx-console>)
- Look for the `jboss.system:type=ServerInfo` MBean and click on the link
- On the page that comes up, look for the `listThreadDump` operation and click on the `Invoke` button

Using Twiddle:

- From the command prompt, `cd` to `JBOSS_HOME/bin` folder
- Run the following command:
 - `./twiddle.sh invoke "jboss.system:type=ServerInfo" listThreadDump > threads.html`
 - Note: Use `twiddle.bat` for Windows OS

This command will generate the thread dump and redirect the output to `threads.html` (you can redirect it to any file of your choice)

It's best to generate multiple thread dumps between a span of few seconds and compare those thread dumps to find any blocked threads.

Reporting problems

Also, search the community forums to see if someone else is experiencing the same problem. The forums are at link <http://community.jboss.org/>. You can also obtain a support contract via <http://www.jboss.com/services/subscriptions/> and then access <https://www.redhat.com/wapps/sso/jboss/login.html?edirect=http%3A%2F%2Fsupport.redhat.com%2Fjbossnetwork>

Search for existing issues that also report the same problem. Access the JIRA issue search via link <https://jira.jboss.org/jira/secure/IssueNavigator.jspa>

When filling in the JIRA, be as precise as possible when reporting the bug. Include as much information as possible. Include the steps needed to reproduce the problem.

If possible, create a standalone test case that reproduces the bug that can be attached to a JIRA issue. The JIRA issue is more likely to be fixed if a unit test is attached (or at least a test case). Even better is if a solution, in the form of a patch (output of doing `"svn diff > fix.patch"`), is attached.

JBoss EAP 5 VS AS 5

The focus of the JBoss AS project is continuous innovation - fueled by Open Source community collaboration - on the bleeding edge of Java Middleware. Quickly bringing emerging standards and technology into the mainstream through its large user base and active community. This focus

implies continuous change and a rapid release cycle with minor releases every one or two months and major releases every six months. Red Hat does not provide support for JBoss community projects.

Red Hat does provide world-class support, consulting and training for the JBoss Enterprise Platforms. JBoss Enterprise Platforms balance innovation with enterprise class stability by integrating the best of Open Source projects like JBoss AS. The JBoss Enterprise Platforms are certified against a broad range of Operating Systems, Databases and other 3rd party applications and tools; meet very strict industry security standards; are pre-tuned and secured by default so are ready

to support your business critical applications and services.

Red Hat recommends JBoss projects as a place to get involved in shaping the Open Source middleware landscape and as a way to understand how the technology landscape is evolving. Red Hat recommends JBoss Enterprise Platforms for demanding business critical production workloads where security, performance, reliability and long-term, world-class support are imperative.

See this link for information on training <http://www.jboss.com/services/training/>

ABOUT THE AUTHORS



Scott Marlow
Core Engineer on the JBoss AS team. Over 20 years experience building enterprise development software, from database server to developer tools (such as PowerBuilder and four different application servers). Five years experience contributing to JBoss OSS projects (Application Server, Clustering, JGroups, JBoss Cache, Hibernate). Scott enjoys coaching and playing soccer in his spare time.

Jaikiran Pai
Employed at RedHat and is part of the JBoss EJB3 development team. Jaikiran completed his graduation in 2004 and started working in a software company in Pune, India. In his role as a software developer, he was part of projects which involved Java and JavaEE. During this period, he developed interest in JBoss Application Server and started spending his spare time in JBoss community forums. In 2009, Jaikiran was offered a job at RedHat to be part of his favourite project - JBoss Application Server.

Jaikiran can often be found either at the JBoss forums or at his other favourite place <http://www.javaranch.com/>. Occasionally, Jaikiran blogs at <http://www.jaitechwriteups.blogspot.com/>

Shelly McGowan
Member of the JBoss Application Server development team. She has several years of software development experience most recently on Java Enterprise Edition technologies such as EJB and EJB 3 persistence.

Brian Stansberry
My background is in International Business and East Asian Studies, with a B.A. from Michigan State and an M.A. from Stanford. Before getting bitten by the software bug, I had a successful career in corporate finance in the semiconductor industry. Part of that oddly enough involved web application and other types of software development. But since I realized in the late 1990s that my true interest was in software, not finance, I've focused on server-side development and Java. I started working on JBoss in 2003 and joined the company in 2005. My other main interests are China (I speak Mandarin Chinese and visit China regularly) and hanging out with my family. I live in St. Louis, MO.
Expertise: JBoss AS Clustering JBoss AS in general JBoss Cache PojoCache JGroups mod_cluster
Occupation: Lead, JBoss AS Clustering

Len DiMaggio
JBoss middleware QE engineer and team lead. Len is a frequent contributor to JBoss blogs and DZone (<http://soa.dzone.com/users/ldimaggio>).

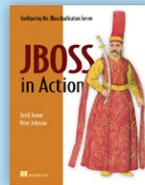
RECOMMENDED BOOK



This book will kick-start your productivity and help you to master JBoss AS development. The author's experience with JBoss enables him to share insights on JBoss AS development, in a clear and friendly way. By the end of the book, you will have the confidence to apply all the newest programming techniques to your JBoss applications.

BUY NOW

books.dzone.com/books/jboss-5-development



JBoss in Action is the first book to focus on teaching readers in detail how to use the JBoss application server. Unlike other titles about JBoss, the authors of JBoss in Action go deeper into the advanced features and configuration of the server. In particular, it focuses on enterprise-class topics, such as high availability, security, and performance.

BUY NOW

books.dzone.com/books/jboss-action-configuring-jboss

Browse our collection of over 95 Free Cheat Sheets



Free PDF

Upcoming Refcardz

- Java EE Security
- Adobe Flash Catalyst
- Network Security
- Subversion



DZone communities deliver over 6 million pages each month to more than 3.3 million software developers, architects and decision makers. DZone offers something for everyone, including news, tutorials, cheat sheets, blogs, feature articles, source code and more. **"DZone is a developer's dream,"** says PC Magazine.

DZone, Inc.
140 Preston Executive Dr.
Suite 100
Cary, NC 27513
888.678.0399
919.678.0300

Refcardz Feedback Welcome
refcardz@dzone.com
Sponsorship Opportunities
sales@dzone.com

