**WildFly(JBoss) vs. Tomcat**

Both WildFly and Tomcat are *Java application servers*, but they have difference strength. Making the wrong choice can result in more work than necessary.

The **JBoss** application server (aka *JBoss AS*) is an **application server** based on Java. It is an open source software server and is usable in any operating system supported Java (because the server is Java based).

Apache Tomcat is a **servlet container** (meaning it is a Java class that operates under the strictures of the Java Servlet API – a protocol by which a Java class responds to an http request). This is an open source server, providing a ‘pure Java’ HTTP web server environment in which code written in Java is capable of running.

1. WildFly is a application servers with access to the whole JEE stack while Tomcat is servelet server
2. Developers of complex Java enterprise applications should choose JBoss (or GlassFish), while those who don’t need the full JEE stack are better off with Tomcat plus any extensions they need.
3. JBoss makes use of the Java EE specification; Tomcat makes use of Sun Microsystems specific specifications

**JMX**

Java Management Extensions (**JMX**) is a Java technology that supplies tools for managing and monitoring applications, system objects, devices (such as printers) and service-oriented networks. Those resources are represented by objects called MBeans (for Managed Bean).

**JBoss MBean Services**

As we have seen, JBoss relies on JMX to load in the MBean services that make up a given server instance's personality. All of the bundled functionality provided with the standard JBoss distribution is based on MBeans. The best way to add services to the JBoss server is to write your own JMX MBeans.

There are two classes of MBeans: those that are independent of JBoss services, and those that are dependent on JBoss services. MBeans that are independent of JBoss services are the trivial case. They can be written per the JMX specification and added to a JBoss server by adding an mbean tag to the deploy/user-service.xml file. Writing an MBean that relies on a JBoss service such as naming requires you to follow the JBoss service pattern. The JBoss MBean service pattern consists of a set of life cycle operations that provide state change notifications. The notifications inform an MBean service when it can create, start, stop, and destroy itself. The management of the MBean service life cycle is the responsibility of three JBoss MBeans: SARDeployer, ServiceConfigurator and ServiceController.

**JBOSS INTERVIEW QUESTIONS**

**Question 1. What Is Jboss?**

**Answer :**

* JBoss is a popular open source application server based on JEE technology.
* Being JEE based, the JBoss supports cross-platform java applications.
* It was embedded with Apache Tomcat web server.
* It runs under any JVM of 1.3 or later versions.
* JBoss supports JNDI, Servlet/JSP (Tomcat or Jetty), EJB, JTS/JTA, JCA, JMS, Clustering (JavaGroups), Web Services (Axis), and IIOP integration (JacORB).

**Question 2. What Is Jboss Cache In Short?**

**Answer :** JBoss cache is a product. Frequently accessed Java objects are cached by utilzing JBoss cache to improve the performance of e-business applications. JBoss decreases the network traffic and increases the scalability of applications by eliminating unnecessary database access. Fully transactional features and highly configurable set of options which are to deal with concurrent data access, are provided by JBoss cache in an efficient manner possible for the applications.

**Question 3. What Is Jboss Jbpm?**

**Answer :** JBoss JBPM is a workflow and BPM engine. Enabling the creation of business processes that coordinates between people, applications and services is the functionality of BPM engine. The combination of workflow applications development with process design is a feature of JBoss jBPM. The business process is graphically represented to facilitate a strong link between the business analyst and technical developer. This feature is provided by the JBoss jBPM process designer.

**Question 4. How Do You Monitor Jboss And Detect The Bottleneck Of An Application?**

**Answer :** Different components of the application are to be measured. This step is to find where the degradation is, whether it is external or internal and where is the application spending all the time. Using Jboss JMX agents and monitoring the deployed components to the application server involves in the first step.

After finding the most of the time spent by specific components or libraries or most of the resources, one can use Jprobe a specialized tool for examining the single object or the objects loaded in the memory.

**Question 5. What Is Jta?**

**Answer :**

Java Transaction API (JTA) specifies standard Java interfaces between a transaction manager and the parties involved in a distributed transaction system: the resource manager, the application server, and the transactional applications.

The Javaâ„¢ Transaction API (JTA) allows applications to perform distributed transactions, that is, transactions that access and update data on two or more networked computer resources.

The Java Transaction API consists of three elements: a high-level application transaction demarcation interface, a high-level transaction manager interface intended for an application server, and a standard Java mapping of the X/Open XA protocol intended for a transactional resource manager.

**Question 6. What Is The Difference Between Hibernate And Ejb 3? Do Not You Think Ejb 3 Is Just A Clone Of Hibernate?**

**Answer :**

The perception of EJB3 as being a simple clone of Hibernate is primarily based on developer familiarity with Hibernate and a similarity of naming, as well as common purpose, and that Hibernate is morphing itself into an EJB3 implementation based on the work going into the specification, not the other way around.

EJBs are supposed to be components, in the sense that they’re not just one class, but a set of classes, descriptors and usage and management contracts. All of this in order to allow a container (JBoss, Weblogic, etc.) to provide services to those components, and to be able to reuse and distribute this components. This services are, among others, transactions, concurrent access control, security, instance pooling, etc.

Hibernate is “just” an ORM (Object/Relational Mapping) tool. Quick and dirty, this means you can store an object tree belonging to an class hierarchy in a relational DB without writing a single SQL query. Quite cool, IMO. But no transaction control, no instance pooling, no concurrency control, and certainly no security.

**Question 7. Which Hibernate Object Wraps The Jdbc Connection?**

**Answer :**

The Session interface wraps a JDBC Connection. This interface is a single threaded object which represents a single unit of work with application and persistent database. It’s retrieved by the SessionFactory’s openSession() method

**Question 8. Is The Session Factory Thread Safe?**

**Answer :**

Yes, that is many threads can access it concurrently and request for sessions. It holds cached data that has been read in one unit of work and may be reused in a future unit of work. Good practice is to create it when the application is initialized.

**Question 9. How Can You Start A Jta Transaction From A Servlet Deployed On Jboss?**

**Answer :**

JBoss registers in the JNDI tree a JTA UserTransaction Object which can be user to manage a distributed transaction.

**Question 10. What If You Need To Span Your Transaction Across Multiple Servlet Invocations?**

**Answer :**

You can’t with a Servlet. A JTA transaction must start and finish within a single invocation (of the service() method). You should consider using a Stateful SB. In a SFSB with a JTA transaction, the association between the bean instance and the transaction is retained across multiple client calls.

**Question 11. What Are The Differences Between Ejb 3.0 And Ejb 2.0?**

**Answer :**

EJBs are now plain old Java objects (POJO) that expose regular business interfaces (POJI), and there is no requirement for home interfaces.

* Use of metadata annotations, an extensible, metadata-driven, attribute-oriented framework that is used to generate Java code or XML deployment descriptors.
* Removal of the requirement for specific interfaces and deployment descriptors (deployment descriptor information can be replaced by annotations).
* Interceptor facility to invoke user methods at the invocation of business methods or at life cycle events.
* Default values are used whenever possible.
* Reduction in the requirements for usage of checked exception.
* A complete new persistence model (based on the JPA standard), that supersedes EJB 2.x entity beans

**Question 12. What Is The Difference Between A Local-tx-datasource And A Xa-datasource? Can You Use Transactions In Both?**

**Answer :**

A local-tx-datasource identifies a data source that uses transactions, even distributed trans actions within the local application server, but doesnt use distributed transactions among multiple application servers. An xa-datasource on the other hand identifies a data source that uses distributed transaction among multiple application servers.

**Question 13. What Do You Need To Set-up A Cluster With Jboss?**

**Answer :**

Basically starting JBoss with the configuration contains everything needed for clustering:

**It has all the libraries for clustering:**

* JGroups.jar, jboss-cache.jar.
* Clustered beans (cluster-service.xml).
* HA-JNDI.
* HTTP session replications (tc5-cluster-service.xml).
* Farming.
* HA-JMS.

**Question 14. What Optimization Could I Use If The Ejb Container Is The Only Point Of Write Access To The Database?**

**Answer :**

You could activate the “Commit Option A” that is the container caches entity bean state between transactions. This option assumes that the container has exclusive access to the persistent store and therefore it doesnt need to synchronize the in-memory bean state from the persistent store at the beginning of each transaction.

**Question 15. Which Component Handles Cluster Communication In Jboss?**

**Answer :**

The JGroups framework provides services to enable peer-to-peer communications between nodes in a cluster. It is built on top a stack of network communication protocols that provide transport, discovery, reliability and failure detection, and cluster membership management services.

**Question 16. Is It Possible To Put A Jboss Server Instance Into Multiple Cluster At The Same Time?**

**Answer :**

It is technically possible to put a JBoss server instance into multiple clusters at the same time, this practice is generally not recommended, as it increases the management complexity.

**Question 17. What Do You Know About Seam?**

**Answer :**

Built on the standards Java Server Faces and EJB 3.0, JBoss Seam unifies component and programming models and delivers a consistent and powerful framework for rapid creation of web applications with Java EE 5.0. Seam simplifies web application development and enables new functionality that was difficult to implement by hand before, such as stateful conversations, multi-window operation, and handling concurrent fine-grained AJAX requests. Seam also unifies and integrates popular open source technologies like Facelets, Hibernate, iText, and Lucene.

**Question 18. Does Seam Run On Other Application Servers Besides Jboss?**

**Answer :**

Seam runs beautifully on other application servers – just like everything else the Hibernate team does, this is not a JBoss-only thing.

**Question 19. Which Jdk Is Needed To Run Seam?**

**Answer :**

Seam only works on JDK 5.0 and above. It uses annotations and other JDK 5.0 features.

**Question 20. How Would You Convince My It Department To Adopt Soa?**

**Answer :**

In my opinion one of the biggest obstacle in the movement towards SOA adoption is the organization own IT department.Too many people in the IT organization conceive SOA as a technology concept only, and as such think of SOA as just a set of technologies and infrastructure for exposing, securing, running, and managing Services.

Put it this way, SOA is nothing more than Web Services and standardized middleware. The critical flaw in thinking is confusing the technology that sits beneath the Services level of abstraction and the mechanism by which Services are accessed with the architectural approach that aims to decouple the implementation from the consumption and focus on sustainable architecture that allows for continuous change.

Successful SOA adoption requires a cultural shift in the way IT is done. The Service-oriented movement to agility and loose coupling demands a shift from traditional, waterfall styles of development (design-build-test-deploy-manage) to iterative approaches to continuous Service modeling

**Question 21. What Do You Think About Bpel And Bpm ? How Do They Compare?**

**Answer :**

* + BPEL and BPM are quite different things so they cannot even be compared. The problems boils down to the fact that these years maybe BPEL has been marketed for something which isn’t: that is a Business Process Management framework.
  + BPEL is made up for service orchestration, that is publishing new services as a function of other services.
  + while BPM si needed for handling human task management functionalities and subprocess management.

**Question 22. What Is The Difference Between Jax–ws And Jax-rpc?**

**Answer :**

Java API for XML-Based RPC (JAX-RPC) is a Legacy Web Services Java API, it uses SOAP and HTTP to do RPCs over the network and enables building of Web services and Web applications based on the SOAP 1.1 specification, Java SE 1.4 or lower.JAX-WS 2.0 is the successor to JAX-RPC 1.1. JAX-WS still supports SOAP 1.1 over HTTP 1.1, so interoperability will not be affected. However there are lots of differences:

* + JAX-WS maps to Java 5.0 and relies on many of the features new in Java 5.0 like Web Service annotations.
  + JAX-RPC has its own data mapping model, JAX-WS’s data mapping model is JAXB. JAXB promises mappings for all XML schemas.
  + JAX-WS introduces message-oriented functionality, dynamic asynchronous functionality which are missing in JAX-RPC.
  + JAX-WS also add support, via JAXB, for MTOM, the new attachment specification.

**Question 23. Do You Know How You Could Add Support For Web Service Transactions?**

**Answer :**

JBossTS supports Web Services transactions, including extended transaction models designed specifically for loosely-coupled, long running business processes. J2EE transactions can integrate seamlessly with Web Services transactions using our integrated, bi-directional transaction bridge. Interoperability with many other vendors is provided out-of-the-box and JBoss is an active participant in these standards.

**Question 24. What Version Of Jboss As Do I Need To Run Seam?**

**Answer :**

For Seam 1.3: Seam was developed against JBoss 4.2. Seam can still be run against JBoss 4.0. The seam documentation contains instructions for configuring JBoss 4.0.

For Seam 1.2: Since Seam requires the latest edition of EJB3, you need to install JBoss AS from the latest JEMS installer. Make sure that you select the “ejb3” or “ejb3+clustering” profile to include EJB3 support. Also, the jboss-seam.jar library file from the Seam distribution must be included in each Seam application you deploy.

**Question 25. Can I Run Seam Outside Of Jboss As?**

**Answer :**

Yes, you can run Seam applications in plain Tomcat 5.5+ or in the Sun GlassFish application server. To run Seam application in Tomcat, you need a number of additional library files and a few configuration files to bootstrap the JBoss EJB3 inside Tomcat.

**Question 26. Can I Run Seam In A J2ee Environment?**

**Answer :**

Yes, as of Seam 1.1, you can use Seam in any J2EE application server, with one caveat: you will not be able to use EJB 3.0 session beans. However, you can use either Hibernate or JPA for persistence, and you can use Seam JavaBean components instead of session beans.

**Question 27. Can I Run Seam With Jdk 1.4 And Earlier?**

**Answer :**

No, Seam only works on JDK 5.0 and above. It uses annotations and other JDK 5.0 features.

**Question 28. Where Can I Find Seam Examples And Documentation?**

**Answer :**

The source code and build script of all Seam example applications are included in the examples directory of the Seam distribution.

**Question 29. Is It True That Seam Only Works With Jsf?**

**Answer :**

Seam only supports JSF as a view framework at this time. We plan to support other web frameworks in the future. We like JSF because it is a component-based UI framework, which fits really well with Seam’s component-based approach to business objects and persistence objects. Seam made a major improvement to JSF by eliminating almost all XML configuration for backing beans — you can now define back beans from POJOs or EJB3 components using simple annotations.

We recommend you use Facelets, instead of JSP, with JSF. Facelets provide a powerful templating framework, better appplication performance, and allows us to write much simpler JSF pages. Please see the Seam booking example application for an example on how to use Facelets.

**Question 30. Can I Use Ajax With Seam?**

**Answer :**

Yes, Seam provides excellent support for AJAX. First, Seam supports the ICEfaces and Ajax4JSF Ajax component libraries for JSF. If you prefer a more “old fashioned” approach, Seam provides a complete JavaScript remoting framework which lets you call Seam components and subscribe to JMS topics directly from the client.

**Question 31. Can I Unit Test Seam Applications Without Starting The Application Server?**

**Answer :**

Yes, Seam provides its own integration test framework based on TestNG. You can easily mock all Seam services using those facilities without ever loading an application server or a database.

**Question 32. How Do You Perform A Hot Deployment?**

**Answer :**

**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

**Question 33. How Do You Enable/disable A Hot Deployment?**

**Answer :**

* + To enable, put file hdscanner-jboss-beans.xml in  JBOSS\_HOME/server/<profilename>/deploy
  + To disable hot deployment, remove the hdscanner-jboss-beans. xml from the deploy folder or rename it to hdscanner-jbossbeans. xml.bak (.bak files are ignored).

**Question 34. What Is Jboss Eap?**

**Answer :**

JBoss Enterprise Application Platform is an open source implementation of the Java EE suite of services

**Question 35. How Do You Deploy An Application To Jboss?**

**Answer :**

 Choose the server profile to which you want to deploy the application

* + 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.

**Question 36. What Does Mgmt-user.properties Contain?**

**Answer :**

All admin console users and password (encrypted) are stored in mgmt.-user.properties file.

**Question 37. What Are The Important Types Available For Marker File Deployment?**

**Answer :**

* + .dodeploy – instruct to deploy
  + .deployed – indicate the file is deployed
  + .pending – deployment is still pending
  + .undeployed – confirmation that application is undeployed
  + .failed – deployment is failed for some reason
  + .skipdeploy – instruct JBoss to ignore the files for auto-deployment

**Question 38. What Marker File Type Is Required To Instruct Jboss To Deploy?**

**Answer :**

.dodeploy file suffix is needed for JBoss to deploy or redeploy an application.  
For ex: **myfirstapplication.war.dpdeploy**

**Question 39. How Can You Deploy An Application?**

**Answer :**

There are three possible ways to deploy an application in JBoss application server.

* + **Admin Console** – you can deploy the necessary application files through the administration console.
  + **Auto-deploy** – leverage file system deployment scanner to auto deploy files from deployments folder.
  + **Automation** – use automation tool/ant/scripting to deploy an application.

**Question 40. What Are The File Types You Can Deploy In Jboss?**

**Answer :**

You can deploy almost any kind of Java/J2EE application, and it supports the following file format.

* + **WAR:** Web application archive
  + **SAR:**Service archive
  + **JAR:** Java Archive
  + **EAR:** Enterprise application archive

**Question 41. What Module Needed To Integrate Apache With Jboss?**

**Answer :**

There are two modules you can use to connect JBoss with Apache.

* + mod\_proxy.
  + mod\_jk.

**Question 42. What Is Difference Between <validate-on-match> And <background-validation>?**

**Answer :**

* + <validate-on-match> validate the database connection every time, and if a connection is not valid, it will write a warning in the logs.
  + Having “validate-on-match” configured may have a little high load on the database as it may create lots of requests.
  + <background-validation> validate the connection periodically based on what frequency is configured for “background-validation-millis”. The default configuration is set to zero means disabled.
  + Having “ background-validation” set to true will create fewer database connections and it’s side-effects would be not detecting immediately if dead connections.

**Question 43. Can You Create A Cluster In Standalone Mode?**

**Answer :**

Yes, clustering is possible in standalone mode. However, an application must be deployed on each server/JVM in standalone mode.

**Question 44. What Is The Difference Between Standalone And Domain Mode?**

**Answer :**

* + Standalone mode is single JVM process where every JBoss server has its configuration. If you just need one JVM or development environment, then standalone would be perfect.
  + Domain mode may have multiple servers where all configuration is managed centralized and often used in production environment.

**Question 45. How To Increase Java Heap Memory In Jboss 7?**

**Answer :**

Heap Memory can be increased in a respective conf file. To increase memory for standalone;

* + Go to bin folder
  + Edit the standalone.conf file and look for “JAVA\_OPTS=” argument line
  + The default configuration will have minimum 64 MB and maximum 512 MB. You can increase to the desired value.
    - Xms – specify the minimum heap size.
    - Xmx – specify the maximum heap size.

In a similar way, you can adjust the memory for a domain in domain.conf file

**Question 46. How To Start Jboss In Standalone Mode?**

**Answer :**

Go to bin folder where JBoss is installed and start with the following command.

./standalone.sh

**Question 47. What Must Be Done To Access Admin Console?**

**Answer :**

The user must be created under “ManagementRealm” to have console operational. To create the user, you can go to bin folder and execute add-user.sh script.

**Question 48. What’s The Default Port To Access Administration Console In Jboss 7?**

**Answer :**

9990 is the default port. If it’s installed on server1 then you need to access like:  
http://server1:9990/admin-console

**Question 49. How To Install Jboss On Linux Server?**

**Answer :**

JBoss installation is very straightforward. You need to download the desired version from JBoss official download page in zip or gz format.

Once downloaded, just extract the file to the location you want to install. If you downloaded zip format, then you can use unzip command to extract it.

unzip jboss-as-7.1.1.Final.zip

**Question 50. Which Component Is Responsible For Handling Clustering?**

**Answer :**

JBoss clustering is on top of JGroups toolkit which helps to create, delete, membership detection, notification, etc. in the cluster.

**Question 51. What Are The Logging Levels Available?**

**Answer :**

**There is five possible level:**

* + FATAL
  + ERROR
  + WARN
  + INFO
  + DEBUG

**Question 52. What Is The Directory Structure In Jboss?**

**Answer :**

The following directories are available after JBoss is installed.

* + modules
  + bundles
  + domain
  + standalone
  + appclient
  + bin
  + docs
  + welcome-content

**Question 53. What Is A .war, .ear Or A .jar File??**

**Answer :**

* + WAR – web application archive is a jar file used to distribute a collection of JavaServer Pages, Java Servlets, Java classes, xml files, tag libraries, static web pages (html & others) and others resources that constitute a web application.
  + JAR – Java Archive, is an archive file format typically used to aggregate many Java class files and associated metadata and resources into one file to distribute application software, libs on the Java platform.
  + EAR – Enterprise Archive is used to package one or more moduels into a single archive so the deployment of the various modules into app server happen simultaneiously and coherently.

**Question 54. What’s The Difference Between A Tar File And A War File?**

**Answer :**

* + TAR – derived from tape archive is a file format to archive bitstreams and name of a program to handle such files.  It is commonly used to collect many files into one large file for distributing or archiving while preserving the filesystem info.
  + WAR – web application archive is a jar file used to distribute a collection of JavaServer Pages, Java Servlets, Java classes, xml files, tag libraries, static web pages (html & others) and others resources that constitute a web application.

**Question 55. How Do You Look Inside A Jar File?**

**Answer :**

 jar –tf <filename.jar>