

COMPARING GLASSFISH™ AND JBOSS WHICH APPLICATION SERVER IS RIGHT FOR YOU?

White Paper
October 2009

Abstract

The choice of an application server depends on many variables—from price, performance, features, and ease of use to the quality of documentation and support. No single offering can claim to be the optimal choice for every project or every development team. However, the majority of developers building Java applications will find that the GlassFish™ application server offers distinct advantages compared with JBoss. This paper summarizes the features and capabilities that make GlassFish a superior choice for building, deploying, and managing enterprise-class Java applications and Web services.

Table of Contents

Executive Summary	3
Overview: GlassFish and JBoss Application Servers.....	4
GlassFish Advantages vs. JBoss	6
Java EE Certification	6
Performance and Price	6
Ease of Use and Administration.....	7
High Availability and Clustering	8
Scripting/Dynamic Frameworks	9
Web Services	9
IDE Support/Tooling.....	9
Integration/Connectivity	10
Technical Support and Documentation	10
Upgrade/Update Tools	11
GlassFish Community and Product Road Maps	11
GlassFish Momentum.....	11
Conclusion	13
For More Information	13

Executive Summary

If applications are crucial to your business, then the selection of an application server is a strategic business decision. And from a business perspective, GlassFish is a superior alternative to JBoss for a broad range of organizations and development teams.

Sun recognizes that personal preferences, legacy investments, and even corporate mandates can play a role in the selection of an application server, and that no single solution meets every conceivable need. However, based on the comprehensive capabilities offered by GlassFish and feedback received from both the GlassFish and JBoss development communities, Sun is confident in asserting that GlassFish offers compelling advantages for enterprise-class Java applications and Web services.

GlassFish is the only open source application server that combines Java™ Enterprise Edition 5 (Java EE 5) certification, exhaustive and accessible documentation, intuitive administration and configuration features, five-nines availability, interoperability with Microsoft Windows, feature-rich integration with NetBeans and Eclipse, and alignment with Solaris™, OpenSolaris™, and MySQL™ —along with responsive support at an affordable price.

Overview: GlassFish and JBoss Application Servers

GlassFish is the open source application server delivered by the GlassFish Community, which was originally launched by Sun in 2005. GlassFish v2 has notched more than ten million downloads and 250,000 product registrations in the past 12 months. In addition, GlassFish is targeted by open source projects more than any other application server, according to a recent report by Ohloh.¹

GlassFish v3, the new version scheduled for release in the near future, features an architecture that is modular by default, allows for rapid, iterative Java Web development, and supports dynamic languages. It is an excellent platform for deploying rich Internet applications backed by Java or dynamic languages such as jRuby, Groovy and Jython.

Sun™ GlassFish Enterprise Server is the commercially supported Java™ Enterprise Edition (EE) 5 compatible offering. The commercial version provides up to 24x7x365 phone and email support, product patches, and additional tools that improve monitoring and boost performance in a production environment.

Sun™ GlassFish Portfolio further expands the options available to developers by integrating the Sun™ GlassFish Enterprise Service Bus (ESB), the Sun™ GlassFish Web Space Server, and the Sun™ GlassFish Web Stack into the commercial platform called Sun GlassFish Portfolio.

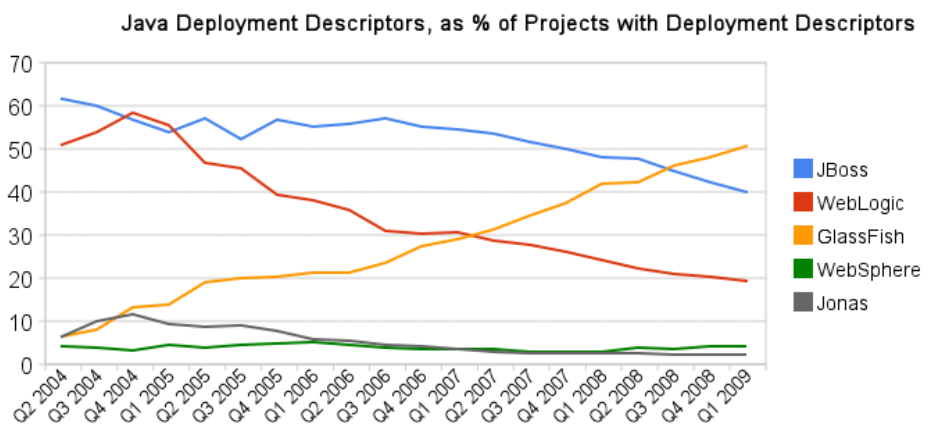


Figure 1: Of Java projects that target Java EE, GlassFish is now the leading application server, with just over 50% of projects targeting GlassFish, according to a report by Ohloh (recently acquired by SourceForge).

¹ "Application Server Trends among Open Source Developers," available online at: http://www.sun.com/software/products/glassfish_portfolio/Ohloh.pdf

The JBoss application server is a certified Java platform for developing and deploying enterprise applications. It supports both traditional APIs and Java EE APIs and includes improved performance and availability through buddy replication. It integrates Apache Tomcat as its Web container (the part of an app server that manages servlets, JavaServer Pages, and other Web-tier components), and includes capabilities for data caching, clustering, messaging, transactions, and an integrated Web services stack.

GlassFish Advantages vs. JBoss

Highlighting the differences between GlassFish and JBoss is not in any way meant to impugn the quality of JBoss or the JBoss Community. However, it is important for developers to be aware of the trade-offs that may be implicit in the choices they make. Below are a few examples of specific capabilities and features where GlassFish holds an advantage over JBoss, and these advantages should be considered by developers who are building, deploying, and managing enterprise-class Java applications and Web services.

Java EE Certification

- GlassFish was the first Java EE 5 certified application server, and currently holds a three-year lead over JBoss in delivering a Java EE 5 certified commercial product. Simply put, GlassFish does Java EE 5 really well, right now, with a mature, fully compliant implementation. At the current time, it has been 12 months since JBoss 5 Community shipped and there is still no supported JBoss 5 EAP (Enterprise Application Platform). GlassFish ships the community and supported versions simultaneously—no need to wait, no need to migrate. Moreover, Sun has already supported two full architectural cycles (v1 and v2) of GlassFish and is committed to supporting v3, providing a level of support experience JBoss cannot claim.
- **GlassFish is aiming to be the first Java EE 6 certified application server**, giving developers a head start in taking advantage of the latest advancements in Java technology. For example, GlassFish v3 Preview offers early access to Java EE 6 technologies and the new Java EE 6 Web Profile SDK Preview. Developers building traditional enterprise applications have access to the rich features of the full Java EE 6 platform, and developers building Web applications targeting the Web container can use the Java EE 6 Web Profile. Because GlassFish v3 Preview uses a microkernel architecture based on OSGi, developers can begin with the Java EE 6 Web Profile and use the Update Center to dynamically upgrade to the full Java EE 6 platform. A Java EE 6 certified version of GlassFish is expected later in calendar year 2009; no timetable for Java EE 6 certification has been publicly announced by JBoss.

Performance and Price

- GlassFish is the only open source application server proven to outperform proprietary application servers, as evidenced by SpecJAppServer 2004 benchmark results, making it the fastest open source application server available. For example, on identical Sun™ Fire T2000 server configurations, GlassFish v2 outperformed WebSphere by 43%.² JBoss has never submitted SpecJAppServer 2004 benchmark result, leaving its overall performance in question.

² http://blogs.sun.com/nazrul/entry/glassfish_v2_the_fastest_application

- The Startup Performance Time (SPT) measured by Sun for GlassFish v3 Preview is 7.56 seconds, and the SPT for GlassFish v2.1 is 22.1 seconds. By comparison, the SPT for JBoss was 73.6 seconds.
- The performance of GlassFish continues to improve as new features are introduced. For example:
 - The high-performance, highly scalable Grizzly connector in both GlassFish v2 and GlassFish v3 Preview improves request/response throughput.
 - GlassFish works well with JSR 199, the Java Compiler API, for JSP compilations. Preliminary performance measurements show an order of magnitude improvement in raw Javac compilation speed, and a 3.5x improvement in overall execution when running JSP TCK tests.
- Pricing for commercial, supported versions of GlassFish and JBoss is competitive in most instances, but GlassFish holds a price/performance advantage with proven performance results. The GlassFish ROI calculator³ can help you determine the costs for your environment.

Ease of Use and Administration

- GlassFish is consistently rated highly by developers for its ease-of-use and administration features. Just a few examples:
 - GlassFish v2 and GlassFish v3 Preview both provide a **task-oriented administration console** and configuration wizards that simplify routine administrative chores. With JBoss administrative tasks are performed via JMX Beans. Even with Embedded JOPR in JBoss 5, the feature richness and out-of-the-box functionality does not approach that of GlassFish.
 - While both GlassFish v3 and JBoss provide a standards-based monitoring API, only GlassFish v3 supports lightweight and non-intrusive monitoring, superb integration with DTrace on the Solaris and Macintosh operating systems, and client-side scripting binding for monitoring on all platforms.
 - The **GlassFish Update Center** (available in GlassFish v2 and GlassFish v3 Preview) offers point-and-click access to additional features that can extend GlassFish capabilities. Examples include Open ESB for building loosely coupled applications, Jersey for building RESTful Web Services, and dynamic languages such as JRuby, Jython, and Groovy.
 - The ability to retain sessions across application redeployments (available in GlassFish v3 Preview) is a time saver for developers creating Java Web applications.
- The administrative GUI and CLI of GlassFish are often viewed as superior to JBoss for getting developers up to speed quickly. With GlassFish v3, everything that can be done with the GUI can be done with the CLI, and the CLI is useful for automating configuration and management using provisioning tools. By contrast, ease of use has sometimes been cited as an issue by JBoss users, particularly newcomers.

³ For details see <https://sun.emktg.info/emarketing/html/glassfish/calc/?intcmp=3045>

- GlassFish Enterprise Server provides full SNMP support by implementing the entire J2EE MIB defined by JSR 77.

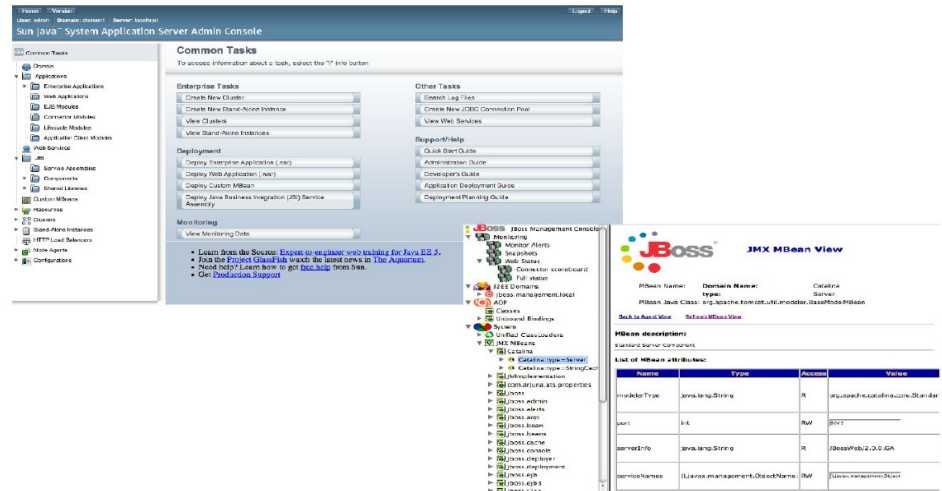


Figure 2: The GlassFish application server features an intuitive admin console that saves time on routine administrative tasks. For many developers, the JBoss interface is more difficult to navigate.

High Availability and Clustering

- GlassFish provides integrated clustering support and sophisticated high-availability capabilities that enable Java applications to meet stringent, enterprise-class service-level agreements. GlassFish Enterprise Server HA supports both in-memory replication for small/medium-sized HA deployments and HADB-based replication for mission-critical deployments. While JBoss supports clustering services, it does not currently support HADB-based replication, consistent hashing-based replication, or SIP replication. GlassFish remains the only open source application server capable of supporting “five nines” availability requirements.
- GlassFish Clustering also supports Automated Delegated Transaction Recovery by a surviving instance when a failure occurs; Timer Migration on failure or stoppage of an instance; In-memory and I/O failover under dynamic shape change of the cluster; and load balancing with Apache and mod_jk — features JBoss does not currently offer.
- GlassFish v2 provides the ability to centrally manage a high-availability cluster and load balancer agent configuration out-of-the-box, a capability JBoss does not currently offer.

Scripting/Dynamic Frameworks

- The GlassFish strategy is to support many scripting and dynamic frameworks, and to focus on high-performance runtime and easy deployment (without requiring war packaging). This enables developers to leverage the ease-of-use, maturity and performance of the GlassFish runtime, and provides a consolidated runtime for production deployments.
- GlassFish v3 allows (J)Ruby and Python applications to be deployed “as is” without a servlet container; no additional packaging is required. Popular Web frameworks such as Django, Rails, and Grails all run on GlassFish. For (J)Ruby, GlassFish supports native deployment and DTrace monitoring in v3; JBoss does not provide these capabilities. For Jython/Django, GlassFish provides WSGI support, native deployment and DTrace monitoring in v3; JBoss does not provide these capabilities.
- Tooling is available on GlassFish for Ruby on Rails (development, deployment, and debugging), but not with the current version of JBoss.

Web Services

- GlassFish v2 and GlassFish v3 Preview provide the Metro stack consisting of JAX-WS, JAXB, and WSIT, enabling developers to create and deploy secure, reliable, transactional, interoperable Web services and clients.
- Sun and Microsoft personnel meet regularly to deliver the most secure and high-performing Web services stack for interoperating between .NET and Java EE with GlassFish, making GlassFish the best choice for interoperating with Microsoft .NET.

IDE Support/Tooling

- Unlike JBoss, GlassFish provides pre-configured NetBeans and Eclipse IDEs, which delivers an integrated experience for both. For example NetBeans 6.7 supports GlassFish v2.1 and GlassFish v3 Promoted builds, with complete Java EE 5 API support via easy-to-use wizards for a great out-of-the-box experience with NetBeans. NetBeans 6.8 and GlassFish v3 will provide an out-of-the-box, integrated Java EE 6 development experience (scheduled for release in calendar year 2009). A stand-alone GlassFish plug-in for the Eclipse IDE is available via the Update Center, and a GlassFish Tools Bundle for Eclipse 1.1 is also available. IntelliJ IDEA 7 and later also includes a GlassFish plug-in.
- GlassFish supports remote deployment, per-project resource configuration, and JAX-WS support. JBoss does not provide these capabilities.

Integration/Connectivity

- The integration of GlassFish, Java EE 5, NetBeans/Eclipse, EJB3, and the JPA enables applications to simply run better and keeps them easier to maintain.
- GlassFish aligns and/or integrates with industry-leading products based on open source such as the Solaris and OpenSolaris operating systems and the MySQL database, further streamlining development efforts. For example, GlassFish already includes integration with the Service Management Facility, and GlassFish v3 Preview includes DTrace probes. GlassFish v3 Preview ships with the MySQL JDBC driver; in fact, GlassFish also ships with and supports the Java Developer Kit (JDK). In fact, although JDK 5 has entered end-of-life, GlassFish Enterprise Server—with a commercial subscription—continues to offer support and patches for JDK 5 that affect the operation of a GlassFish deployment. The net result: with GlassFish there is no finger-pointing among multiple vendors.
- GlassFish provides default integration with Open MQ but can also work with other JMS providers through the GenericJMSRA, a Java EE Connector Architecture-compliant, feature-rich, open source resource adapter to connect to Message-Oriented Middleware (MOM). JBoss does not provide third-party JMS integration.
- GlassFish provides out-of-the-box GlassFish/MQ cluster setup, support for in-process brokers, shared AS/MQ high-availability configuration, three modes of MQ lifecycle management, and a single configuration file for AS and MQ. JBoss does not provide any of these capabilities.
- GlassFish provides extensive configuration support (CLI/GUI/MBeans), highly configurable parameters to tune connection pools for optimal resource utilization, and extensive monitoring support for connection pools. JBoss provides only partial support for these capabilities.

Technical Support and Documentation

- Sun provides a wide range of technical support options for GlassFish, including live call transfer, online support, and developer technical support, enabling developers to take full advantage of GlassFish capabilities and keep systems tuned for maximum performance. GlassFish offers more flexible price-listed options, such as per-CPU, per-server, and “unlimited deployment” options, whereas JBoss historically has price-listed per-CPU only. Sun can also provide service plans, training programs, developer assistance, and consulting services to help you architect, implement and manage a GlassFish application server-based solution, and managed service offerings as an additional option.
- Sun’s subscription pricing is clearly presented in public Web sites. JBoss has removed pricing from its public Web site, suggesting a change in price.

- Extensive documentation exists for both GlassFish and JBoss; however GlassFish provides online searches within a document, online searches in a document set, and deep cross-document links—features that are not currently available for JBoss documentation.
- More than 20 technical documents are available online detailing the functionality of various GlassFish features and capabilities, offering advice about performance tuning, and providing best practices—all written by the Sun documentation team. Much of the available JBoss documentation is written by community members, and varies in its level of detail, tone, and overall usefulness.

Upgrade/Update Tools

- Both GUI-based and CLI-based upgrade tools are available for GlassFish with full documentation. JBoss does not provide comparable tools.
- The industry-strength, IPS-based Update Center provides an easy way to download and install additional features. Product distribution is repository-based, which allows for third-party contributions and leverages the open source community. JBoss does not allow for such contributions.

GlassFish Community and Product Road Maps

- Both GlassFish and JBoss are driven by dynamic, thriving communities of developers. For the GlassFish community, however, code adoption by the larger development community is a specific and fundamental objective. The GlassFish Community is committed to making it easier for other development teams to deliver compliant implementations, thereby accelerating the adoption of Java standards—which is good for the Java Platform and for everyone who is creating and delivering products for that platform.
- Another distinct advantage of the GlassFish Community is its openness about product roadmaps. Exact, specific delivery dates are publicly posted, allowing others to plan their product roadmaps accordingly. ISVs can confidently align their final release plans with those of GlassFish; JBoss does not advertise its roadmaps with dates, nor has it historically delivered on its roadmaps.

GlassFish Momentum

- The rapid acceleration in adoption of GlassFish appears to be coming at the expense of JBoss. GlassFish is now the application server of choice for 73% of new Java EE projects, according to a recent report by Ohloh.
- The Ohloh report also states that GlassFish has become “the market leading application server among open source developers today... GlassFish is well-positioned for future dominance in the market place with a very commanding position among new projects coming on line today.”

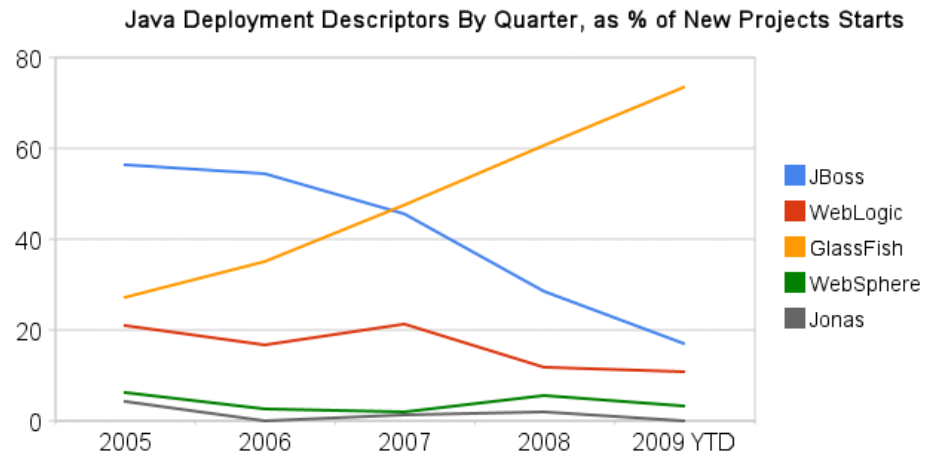


Figure 3: Of Java projects that target Java EE, GlassFish is the leading application server for new project starts with 73% of all Java EE projects, according to a report by Ohloh.

Conclusion

GlassFish combines enterprise-class capabilities with the affordability and flexibility of the open source model. It is the only open source application server that combines Java EE 5 certification, exhaustive and accessible documentation, intuitive administration and configuration features, five-nines availability, feature-rich integration with NetBeans and Eclipse, and alignment with Solaris, OpenSolaris, and MySQL—along with responsive support at an affordable price.

The choice of an application server is highly strategic for businesses, and GlassFish has proven to be a superior strategic option for developers of next-generation Java applications.

For More Information

For additional details about GlassFish, visit

[*http://www.sun.com/glassfish*](http://www.sun.com/glassfish)

To download GlassFish software, visit

[*http://java.sun.com/javaee/downloads/index.jsp*](http://java.sun.com/javaee/downloads/index.jsp)

To join the GlassFish Community, visit

[*https://glassfish.dev.java.net/public/devindex.html*](https://glassfish.dev.java.net/public/devindex.html)

For details about GlassFish Portfolio, visit

[*http://www.sun.com/software/products/glassfish_portfolio/index.jsp*](http://www.sun.com/software/products/glassfish_portfolio/index.jsp)

For success stories and customer ratings, visit

[*http://www.sun.com/software/products/appsrvr/perspectives.xml*](http://www.sun.com/software/products/appsrvr/perspectives.xml)

To read the Ohloh report, visit

[*http://www.sun.com/software/products/glassfish_portfolio/Ohloh.pdf*](http://www.sun.com/software/products/glassfish_portfolio/Ohloh.pdf)



Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA Phone 1-650-960-1300 or 1-800-555-9SUN (9786) Web sun.com

© 2009 Sun Microsystems, Inc. All rights reserved. Sun, Sun Microsystems, the Sun logo, Java, GlassFish, Solaris, OpenSolaris, MySQL, Sun GlassFish Enterprise Server, Sun GlassFish Portfolio, Sun GlassFish Enterprise Service Bus, Sun GlassFish Web Space Server, Sun GlassFish Web Stack, and Sun Fire are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries in the United States and other countries. Information subject to change without notice. Printed in USA 10/09