

# PUTTING OPEN SOURCE TO THE TEST: THE MAKING OF JBOSS ENTERPRISE MIDDLEWARE

---

Through a rigorous productization process, JBoss Enterprise Middleware teams continually harden and transform cutting-edge open source technology into well-tempered enterprise software products with unsurpassed quality, performance, and stability – then deliver it with top-notch support and mission-critical SLAs. Here's how it's done.

June 2008



## TABLE OF CONTENTS

---

ABSTRACT	Page 3
ENTERPRISE IT MEETS OPEN SOURCE: BENEFITS AND CHALLENGES	Page 4
CREATING JBOSS ENTERPRISE MIDDLEWARE	Page 6
Open source for the enterprise	Page 6
Hardening open source software into middleware products	Page 7
Defining products and identifying software sources	Page 7
Branching code and assembling components	Page 8
Defining product release strategy	Page 9
Testing function and quality	Page 9
Certifying platform and database compatibility	Page 10
Validating stack compatibility	Page 11
Meeting requirements for performance, scalability, and resilience	Page 11
Security analysis, testing, and monitoring	Page 11
Platform certification: Behind the scenes	Page 12
Providing enterprise-quality support and service	Page 12
Legal assurance	Page 13
Professional documentation and training	Page 13
Leveraging a reference implementation	Page 13
Software localization	Page 14
Previews of coming attractions	Page 14
JBoss Enterprise Middleware: Delivering the best of both worlds	Page 14
About JBoss Enterprise Middleware	Page 16
About Red Hat	Page 16

## ABSTRACT

---

For enterprise IT organizations and independent software vendors (ISVs), open source software eliminates high up-front license fees, provides transparent access to code, and delivers high-quality capabilities honed by many intelligent minds focusing on a problem. But wholehearted adoption of an open source approach also brings diverse challenges, including highly granular components with disparate versions that must be integrated and tested, variable component stability, uncertain platform and database compatibility, and unknown performance characteristics. Documentation and support may or may not be available to save time for busy developers. While IT managers expect to perform some testing and validation of any new software they adopt, it can be hard to predict the time and effort needed, especially in a complex environment that incorporates multiple platforms and systems. A multitude of open source components – with a multitude of usage agreements – also brings management headaches and legal risks.

JBoss Enterprise Middleware helps companies overcome these and many other challenges by providing the best of both worlds – software developed by a vibrant open source community, with the testing, certification, documentation, training, and support traditionally found only with mature commercial software products. This whitepaper describes how a collection of technology developed by open source projects becomes a set of JBoss Enterprise Middleware products. This productization process – incorporating careful requirements analysis and selection, a stable code branch, a formidable battery of tests, and full documentation – continually delivers products that are hardened, polished and enterprise-ready – stable, scalable, and fully supported. As a result, enterprise IT organizations experience lower middleware TCO and risk, while developers can devote more time to business application challenges.

This paper is intended primarily for senior IT managers responsible for enterprise architecture, application development, and application operations. Managers evaluating middleware solutions will gain a deeper understanding of how JBoss Enterprise Middleware products are developed and why they offer an ideal platform for the development and deployment of mission critical applications and services.

Note: This paper does not provide an overview of the capabilities of JBoss Enterprise Middleware products. If you are new to the products, you may wish to begin with the JBoss Enterprise Middleware data sheet or individual product data sheets available from [www.jboss.com](http://www.jboss.com).





## ENTERPRISE IT MEETS OPEN SOURCE: BENEFITS AND CHALLENGES

---

Open source middleware has been a major boon to IT organizations that want to reduce risk and TCO for their application infrastructures. Today many open source software components power mission-critical applications, including the Linux operating system, Apache Web Server, and JBoss Enterprise Middleware. Open source software offers many advantages, including:

- **Transparent access to code.** With the ability to see how code works, a company's IT staff knows exactly how a vendor has implemented features and industry standards, and has the information needed to address any integration issues quickly and easily. In addition, IT managers can avoid locking their companies into specific vendors or products over the long term.
- **Constant innovation fueled by an organic development process.** Each open source project benefits from a large community of users who download the software, try it out, and provide feedback and suggested changes. Rather than depending on a single in-house team, open source software design is the product of many great minds collaborating across the globe. The pace of innovation is limited only by the amount of time developers can spend.
- **More rational vendor relationships.** Providers of open source software typically earn revenue through support subscriptions renewed on an annual basis, giving them an incentive to provide high-quality support each year and ensure that customers continue to be successful with their software. Because access to new versions is included in the subscription fee, costs are predictable and consistent compared with the up-front purchases that characterize the traditional proprietary software model.
- **Freedom from software license fees.** With free downloads available, open source software eliminates costly software licenses and the cumbersome processes associated with a software purchase. The result is less shelfware and a higher return on investment in application development.

---

**“We’ve been able to attain world-class levels of availability and performance too. Red Hat and JBoss have proven themselves to have enterprise-level tools that, when used well, help us attain what all companies want – great availability, great scale, and at a reduced cost when compared to proprietary solutions.”**

**- Ron Rose, CIO, Priceline**

---



With such compelling advantages, many organizations of all sizes have adopted open source components. But along with these advantages, enterprise IT organizations or ISVs who integrate open source software into their infrastructures also confront some new challenges, including:

- **The need to manage frequent change.** Open source development teams prioritize delivering new capabilities to community members. Their development process creates a constant flow of new ideas and new features, so very active projects may develop and deliver substantial new feature sets in every release—even every minor release. As every software engineer knows, constant change means uncertain stability. The burden of deciding exactly when and how to adopt and test a new version falls on the IT staff, whose needs for stability typically outweigh the need for the new features.
- **Variations in component testing and maturity.** While open source middleware components generally undergo structured functional testing, even the most diligent open source team seldom has the time or resources to perform the full battery of tests needed to deliver robust software that meets stringent enterprise requirements. The open source community's organic testing and feedback process demonstrably can produce very high quality software over time, especially for widely used components. But newer components or those not widely used may receive only very limited testing.
- **Granular software components.** Because most open source software projects are focused to deliver a single component, IT organizations typically need to assemble a number of these components and test them as an integrated solution in order to meet their organization's middleware requirements.
- **Variable access to documentation, training, support, and expertise.** Documentation completeness and accuracy, availability of training, and availability of technical support can vary enormously among open source projects. As a result, IT organizations cannot always be sure of getting the timesaving information and expert help they need. Of course, each open source project creates an ecosystem of interested developers and users who can often provide suggestions or solutions, but this volunteer effort may not meet the needs of enterprise IT staffs supporting production systems.
- **Diverse intellectual property agreements.** Though open source software is delivered without license fees, many companies find the variety of terms and requirements in open source license agreements complex and confusing. The problem is compounded when many open source components are used. To comply with corporate policies and mitigate legal risk, companies must protect their own and others' intellectual property, which requires documenting and managing a sometimes complex array of agreements. In this context, the open source licensing paradigm can present a management challenge.

---

**"We have realized the value of open source for the enterprise, especially in the areas of cost savings, performance, and security."**

**- Barry Strasnick, CIO, CitiStreet**

---

Though each of these challenges can be addressed in a straightforward way, taken together they represent a significant integration, testing, and management burden for the IT organization. Too often, the result is longer implementation cycles, extra risk, and higher than expected cost of ownership. So while open source software can be truly valuable to the organization, it can also produce headaches for IT managers, especially in larger companies using a wide range of platforms and applications.





# CREATING JBOSS ENTERPRISE MIDDLEWARE

---

## OPEN SOURCE FOR THE ENTERPRISE

As open source components have gradually become ubiquitous in some of the largest IT infrastructures in the world, companies that deliver open source software have evolved their business models and internal processes to help both enterprise IT organizations and ISVs take full advantage of the software's benefits while mitigating or eliminating many of the intrinsic challenges. One very important result of this evolution is JBoss Enterprise Middleware, a comprehensive portfolio of certified, enterprise-class open source software products. Enterprise architects can use JBoss Enterprise Middleware to create complete middleware reference architectures that support the entire application lifecycle.

JBoss Enterprise Middleware products, available to customers with support subscriptions, represent the best of both worlds—the dynamic, community-based innovation of open source software, with the stability and support traditionally associated with commercial software products. JBoss Enterprise Middleware eliminates many of the challenges associated with typical open source projects by:

- **Aggregating open source components into well-defined products**, eliminating much of the integration and testing burden of working with granular components and also simplifying software management.
- **Ensuring production-level quality and stability** through a broad and deep testing process, reducing validation time for IT organizations, and minimizing downtime and troubleshooting.
- **Testing for performance and scalability** under enterprise workload conditions.
- **Certifying products on a wide variety of JVM and operating system combinations**, eliminating confusion and extra work for IT organizations.
- **Delivering new capabilities on a predictable, structured release schedule**, insulating companies from the chaos and potential instability associated with constant innovation, while enabling them to absorb new capabilities in an organized way.
- **Localizing software and documentation** for major languages.
- **Guaranteeing support for each new version** (major or minor) for at least five years from release date.
- **Indemnifying companies** against potential intellectual property infringement related to product code.
- **Providing a complete, professional line of documentation and training** for each product to improve development efficiency.
- **Offering a range of technical support SLAs** to meet enterprise IT requirements, along with expertise that IT organizations can call upon when needed.



## HARDENING OPEN SOURCE SOFTWARE INTO MIDDLEWARE PRODUCTS

---

**“Price was certainly a differentiator, but Red Hat’s reliable products and support are prerequisites for considering open source products. As an open source product, the JBoss Enterprise SOA Platform also provides us with the opportunity to bring commonly used features back into the product, reducing in-house development costs.”**

**- Thorbjørn Blixen-Finecke, Chief Architect, Cybercity**

---

The JBoss Enterprise Middleware productization process is a structured sequence of steps through which code from diverse open source projects becomes well defined, fully supported middleware products. While substantial interaction occurs between the teams that manage JBoss.org projects and the teams that deliver JBoss Enterprise Middleware products, these teams have different goals and objectives. The JBoss.org project team is responsible for advancing the state of the art and managing a broad community of contributors, while the JBoss Enterprise Middleware team is charged with delivering a finished software product through a process that typically takes four to five months. The goals of this process are:

- To provide enterprise IT organizations and ISVs with a maximally functional and stable platform on which to develop their applications.
- To eliminate for IT organizations much of the integration work, testing work, and risk traditionally associated with using open source software.
- To provide customers with service level agreements appropriate for 7x24x365 production use.

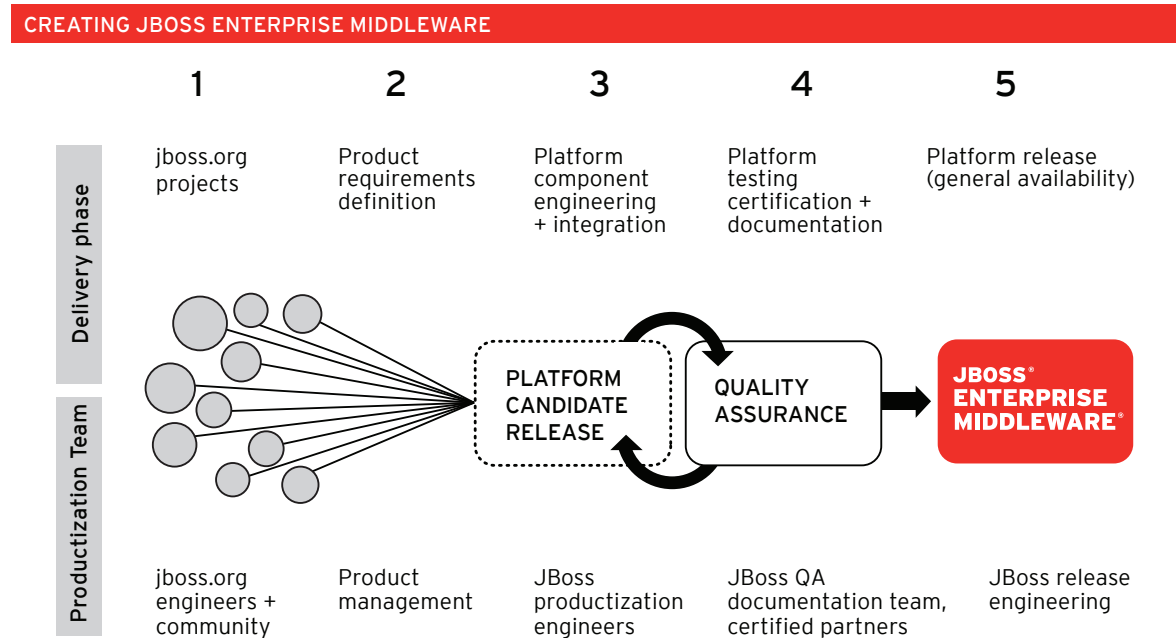
## DEFINING PRODUCTS AND IDENTIFYING SOFTWARE SOURCES

Any product planning process begins with customer and market requirements and a strategy to address them. While JBoss Enterprise product managers gather input from the open source community, they prioritize requirements from enterprise customers and take enterprise IT environments into account to establish the direction of each product individually and the JBoss Enterprise Middleware portfolio as a whole.

The product team’s goal is to provide customers with the best possible products based on state of the art thinking, designs, and code. To analyze the gap between existing capabilities and customer requirements, product managers examine existing JBoss Enterprise Middleware products, recent JBoss.org project developments, and other compatible community software that may be available from open source projects around the world. Special attention is given to components that have become recognized leaders in their categories. For example, application development tools in the JBoss Enterprise Middleware portfolio are all based on Eclipse ([www.eclipse.org](http://www.eclipse.org)) and the JBoss Tools project ([www.jboss.org/tools](http://www.jboss.org/tools)), which also incorporates other open source projects like JBoss RichFaces ([www.jboss.org/jbossrichfaces](http://www.jboss.org/jbossrichfaces)). It also incorporates the run-times for the JBoss Enterprise Application Platform.



When this analysis is complete, the product team identifies the specific components and versions that will be assembled and tested to become the next release of the product. The team then defines specific objectives for the release, balancing ambitious requirements with time and resource constraints. Requirements include not only functional capabilities, but also usability requirements, platform certification, performance characteristics, scalability requirements, installation and migration tools, and other requirements for capabilities that make customers' lives easier.



To create each JBoss Enterprise Middleware product, JBoss productization and delivery teams proceed through a well-defined productization process that takes an average of four to five months to complete.

### BRANCHING CODE AND ASSEMBLING COMPONENTS

Once requirements have been agreed upon and code sources identified, the JBoss Enterprise Middleware productization team creates a code branch that will become the next release of the product. At this point, ongoing open source community development no longer affects the branched code, so the product team can stabilize and test the software without having to incorporate new capabilities. Any code changes to the branch, no matter how minor, are first submitted to the upstream JBoss.org project for approval, and then, once approved, incorporated into the associated Enterprise JBoss Middleware product. The productization team combines the code from these various sources – from JBoss.org projects or other open source projects – and integrates them to form a single enterprise software product.

---

**“We’ve found a high-quality, easily scalable server that can match our business growth, as well as superior professional service from the expert developers behind the technology.”**

**– Michael McDonald, Director of Technology, Continental Airlines**

---

In addition to assembling existing components into an integrated whole, the team may also develop new tools and capabilities where needed to support requirements that are not normally addressed by the open source community, such as migration or diagnostic tools.

### **DEFINING PRODUCT RELEASE STRATEGY**

The JBoss Enterprise Middleware product release schedule allows features and fixes to be delivered to customers in an orderly way while maximizing the stability of their platform. A hierarchy of major releases, minor releases, and cumulative patch releases provides IT organizations with a straightforward, manageable process. Because patch releases are made to the Enterprise branch of the code, customers are insulated from ongoing development in the open source community.

While fixes and changes are made to the Enterprise branch, they are also given to the open source project community, enabling the open source project software to benefit from innovations and patches made by the Enterprise productization team.

When the definition process is begun for the next major release of an Enterprise product, the team begins again by examining the open source community projects, enabling enterprise customers to benefit from the vibrant development process that has been advancing the state of the art. Of course, this new release then undergoes the same productization process to ensure enterprise-level quality, stability, and performance.

### **TESTING FUNCTION AND QUALITY**

The JBoss Enterprise Middleware product testing process delivers production-quality software that represents a safe choice for enterprise IT organizations. While open source projects such as JBoss.org projects each conduct testing before releasing their components to the community for feedback, these tests may not be sufficient to provide the level of quality and stability needed for production use. Community testing may be limited in a variety of ways; depending on developer interest, certain use cases or configurations may be well tested while others are ignored. The JBoss Enterprise Middleware product testing process augments these tests with a wide range of additional component functional tests, integration tests, and compatibility tests designed to raise the bar for quality and stability and allow enterprise IT organizations to use the products with confidence. Wherever possible, real customer cases – test cases provided by customers or discovered during the customer support process – are used in the testing process to ensure that products meet customer expectations.





But product testing does not stop with functional testing. Each JBoss Enterprise Middleware product has a defined set of test criteria that must be passed before a release (major, minor, or patch) is delivered to customers. This process requires testing on a wide range of platforms, configurations, and load conditions as described in the following sections. It also includes installation and management testing.

---

**“The JBoss Enterprise Application Platform was able to deliver the performance that was required and had the professional support necessary to enable us to confidently deploy the software in a production environment.”**

**- Per-Ola Sjöswärd, Executive IT Strategist, Swedish National Police Board**

---

#### CERTIFYING PLATFORM AND DATABASE COMPATIBILITY

For open source projects, the nature and extent of platform testing depends heavily on the platforms available to developers and the time they want to devote to platform testing vs. development of new features. As a result, most open source software is tested only on a single common platform. But developers have long known that differences in operating systems and other platform components can lead to compatibility problems and that software must be tested on the platform or platforms on which it will run in production.

JBoss Enterprise Middleware products are tested with a wide range of Java Virtual Machine (JVM) software and operating system software to ensure that they meet enterprise requirements (see sidebar).

In addition, key middleware integration points such as database connectivity are tested with leading relational database platforms including:

- Microsoft SQL Server
- MySQL
- PostgreSQL
- Oracle
- IBM DB2





## VALIDATING STACK COMPATIBILITY

JBoss Enterprise Middleware products typically undergo diverse types of integration testing in addition to platform and database compatibility testing. First, software components from a number of open source projects are combined to create a single product, and these must be integrated and tested together. Second, the various JBoss Enterprise Middleware products are tested in combination with one another to ensure that the product portfolio operates seamlessly for customers. Third, because JBoss Enterprise Middleware can form the foundation for software products delivered by other vendors (ISVs) – in separate or fully-embedded configurations – these leading vendors also test and validate new JBoss product releases to ensure software stack compatibility. Finally, because middleware always interacts with other software used for building, deploying, and maintaining applications and services, JBoss Enterprise Middleware products are also tested with other popular software tools commonly found in enterprise IT application environments.

## MEETING REQUIREMENTS FOR PERFORMANCE, SCALABILITY, AND RESILIENCE

Traditional open source software components may not be tested under realistic production transaction and data loads. As a result, their performance and scalability characteristics may be unknown. To ensure that the products meet enterprise IT performance and scalability requirements, JBoss Enterprise Middleware products undergo rigorous performance testing including:

- Testing of specific components to identify and eliminate performance bottlenecks and ensure that the product meets established performance criteria.
- Reliability tests of products under high transaction loads for extended periods of time (24-36 hours).

In the past, the JBoss Application Server was frequently used by customers in a clustered environment, but it may not have been tested in that environment before community project release. Today, the JBoss Enterprise Application Platform is tested in clustered environments, and newer failover capabilities are tested in realistic failure conditions, ensuring they will meet customer requirements for continuous application availability in the face of unplanned outages or planned system maintenance.

## SECURITY ANALYSIS, TESTING, AND MONITORING

As part of the JBoss Enterprise Middleware product release planning process, a manifest of all included and dependent technology components is developed for security tracking purposes. Throughout the hardening process and prior to each release (major, minor, or patch), this manifest is reviewed and updated to ensure that any changes with the potential to impact security are evaluated, tested to ensure the product meets security criteria, and appropriately documented. Throughout the five year product support lifecycle, the Red Hat Security Response Team continuously investigates and tracks security issues, categorizes their severity, and ensures that any critical defects are immediately addressed and patched. In essence, the stringent security processes developed to support Red Hat Enterprise Linux are applied to JBoss Enterprise Middleware products in order to deliver middleware that meets and exceeds enterprise security requirements.





#### PLATFORM CERTIFICATION: BEHIND THE SCENES

For an example of the testing difference between JBoss.org projects and JBoss Enterprise Middleware, consider the JBoss Application Server. This component was typically released to the JBoss.org community after it passed two suites of tests – one suite of unit tests and another suite of compatibility matrix tests. While these test suites were substantial, with close to 4,000 tests each, they were run on a single JVM (Sun 1.5 JVM) and on a single platform (Red Hat Enterprise Linux). In contrast, for the JBoss Enterprise Application Platform, which includes JBoss Application Server and over 15 other integrated JBoss.org projects, the enterprise test suites are run against more than 16 different JVM/OS combinations before release, with post-release testing that continues to expand the number of platform combinations. All of the tested and certified combinations are published to ensure that customers adopting JBoss Enterprise Application Platform know exactly which testing has been done and whether the product has been tested and certified for their platforms of choice.

#### SAMPLE LIST OF CERTIFIED PLATFORM COMBINATIONS

BEA JRockit JVM on Red Hat Enterprise Linux 4 x86 and x86\_64

BEA JRockit JVM on Red Hat Enterprise Linux 5 x86 and x86\_64

BEA JRockit JVM on Windows 2003 x86 and x86\_64

HP JVM on HP-UX 11i IA64 and PA-RISC

Sun JVM on Red Hat Enterprise Linux 5 x86 and x86\_64

Sun JVM on Solaris 9

Sun JVM on Solaris 10

Sun JVM on Windows 2003 x86 and x86\_64

Sun JVM on Red Hat Enterprise Linux 4 x86 and x86\_64

#### PROVIDING ENTERPRISE-QUALITY SUPPORT AND SERVICE

It can be difficult for even the best support staff to provide high-quality service for software downloaded from open source community projects. With constantly changing code, many different versions in use, and components that can be combined in a multitude of ways, even highly trained staff may find it quite challenging to replicate a customer's problem or propose a solution.

JBoss Enterprise Middleware products, in contrast, represent stable releases well known to the technical staff who are trained to support those products and versions and can provide both IT organizations and ISVs with the expert help they need. Of course, the best technical support is the one you don't have to use – but even with the high levels of quality provided in JBoss Enterprise Middleware products, software will always require some level of support. To ensure that your middleware support relationship delivers value over the long term, you should select a support organization with proven technical expertise and high levels of customer satisfaction.



## LEGAL ASSURANCE

As IT organizations and ISVs adopt open source technology, they accumulate many different components, each with its own license agreement. To ensure full legal compliance, companies must read and understand their obligations under each agreement, and many do not. In addition, because open source software is created by a broad community of developers, there is increased risk of introducing code that is not standards-compliant or raises intellectual property concerns. At the same time, it is virtually impossible for a large IT organization to scrutinize every piece of code for potential problems. JBoss Enterprise Middleware products eliminate these risks through Red Hat's Open Source Assurance program, which indemnifies and protects customers against any issues arising from product code contents. As a result, organizations know what is in their software and can manage their legal obligations at a higher level.

## PROFESSIONAL DOCUMENTATION AND TRAINING

The quality of documentation and training accompanying a software product can make the difference between a very straightforward project and one with many headaches and wasted hours and days. Documentation for open source project software can vary in quality, accuracy, and completeness – ranging from a few blog entries and forum posts to professional documentation sets. But even the best documentation is seldom provided in localized versions.

The JBoss Enterprise Middleware structured productization process supports the creation of professional documentation sets and training modules that represent valuable, time-saving resources for customers. For each JBoss Enterprise Middleware product, a team of professional technical writers ensures that detailed installation guides, configuration guides, and product release notes are delivered with each product release. To enable products to meet the requirements of a global customer base, these materials are localized for each product release.

### LEVERAGING A REFERENCE IMPLEMENTATION

One important benefit of the JBoss Enterprise Middleware productization approach is that each product becomes a reference standard – identifying which versions of various JBoss.org project components are recommended to interact with each other.

In the case of the JBoss Enterprise Application Platform, more than 15 individual open source project components are evaluated to determine which specific versions should be selected for inclusion in the productization process. Factors such as product maturity, feature exposure, open source project community uptake, and potential for long-term viability are considered. Because selected component versions will undergo the full battery of tests and be fully supported for five years after initial product release, product managers carefully ensure that the selected versions will be able to meet enterprise software requirements.

While component selection and comprehensive product testing are important, added benefit comes from the economies of use that occur when all JBoss Enterprise Middleware users are exercising the same supported reference implementation. Expanded use of a common set of software products results in expansion of customer-driven test cases and usage scenarios, and fosters creation of an enterprise community that can recommend and prioritize features, fixes, and test configurations. The result is products of even higher quality over time.





## SOFTWARE LOCALIZATION

Localization of open source software depends on the specific interests of the community, and quality of translations may vary depending on the resources available. JBoss Enterprise Middleware, in contrast, is professionally localized for major languages.

## PREVIEWS OF COMING ATTRACTIONS

With software from open source projects, mature capabilities are found alongside new, experimental features. For the new user, it can be hard to know which is which. While JBoss Enterprise Middleware products can include newer, experimental features that represent previews of new capabilities, these are clearly identified as Technology Previews that are defined and described in release notes and support policies to ensure that it is always clear to the user which capabilities are backed up by SLAs and which are not. Technology Previews represent features that are likely to be fully supported in a future release but are not yet complete. Customers are encouraged to provide feedback and suggestions. This process enables the JBoss Enterprise Middleware team to engage in a dialog with customers about specific long-term feature requirements and features that are currently under development, while not compromising the stability of more mature capabilities in production use.

## JBoss ENTERPRISE MIDDLEWARE: DELIVERING THE BEST OF BOTH WORLDS

---

**“Our adoption of the JBoss Enterprise SOA Platform, together with Red Hat support, has been extremely successful because it has enabled us to take advantage of the best of both worlds by increasing the reliability of performance levels and by optimizing our IT systems.”**

**- Maurizio Quattrocchi, IT & Telecoms Director,  
National Institute of Design and Mint in Italy**

---

Both IT organizations and ISVs can benefit from adopting JBoss Enterprise Middleware, a suite of robust software products that offer enterprise IT managers substantial value over components from the open source community. These products deliver a robust enterprise application development and deployment platform that eliminates many of the challenges associated with open source middleware while retaining all of its benefits. Products are accompanied by top-notch documentation, training, support, and stringent service level agreements that make the software a safe choice for every organization.

When you choose JBoss Enterprise Middleware, you gain an application architecture that can support your changing requirements over time while insulating you from the disadvantages of rapid innovation. By leaving testing and validation to Red Hat and effectively outsourcing the rationalization and management of your software stack, you substantially lower your cost of ownership, gain higher quality software, and can more quickly deliver applications that support your business strategy and scale to meet future requirements.



For more information about JBoss Enterprise Middleware, visit [www.jboss.com](http://www.jboss.com) or talk with your Red Hat representative.

	SOFTWARE FROM JBOSS.ORG OR OTHER OPEN SOURCE PROJECTS	JBOSS ENTERPRISE MIDDLEWARE PRODUCTS
Software license fee	No	No
Access to source code	Full access	Full access
Availability	Anyone	JBoss Enterprise Middleware subscribers
Development	Open source project community	Open source project community plus JBoss Enterprise Middleware productization team
Enhancements	Ad hoc or experimental software changes depending on the needs of the community	Structured release process includes major and minor releases, patches, and migration tools
Documentation	Varies by project component	Full, professional, localized software documentation
Overall quality	Functional testing and some compatibility testing	Excellent, incorporating a broad array of tests and much more
Long-term stability	Varies by project component due to constant innovation	More stable due to the nature of the productization process. Cumulative patch releases address defects without introducing instability.
Load testing	Limited	Fully load tested
Localization	Depends on community interest and efforts	Professionally localized for major languages
Platform and database compatibility	Generally tested on one popular platform	Certified on more than 16 JVM/ operating system combinations and a range of leading databases
Solution integration	Individual open source components only	Components assembled to create well-defined product. Multiple products tested together to create comprehensive solution
Security coverage	Depends on the interests of the community	Issues identified and addressed by Red Hat's Security Response Team
Training	Varies by component	Full, professional product training available





	SOFTWARE FROM JBOSS.ORG OR OTHER OPEN SOURCE PROJECTS	JBOSS ENTERPRISE MIDDLEWARE PRODUCTS
Support	Mutual support from the community only - no support contracts or SLAs	A range of support subscriptions available depending on your SLA requirements
Support guarantee	None	Support will be available for five years following release date
Legal assurance	None	Customer protected against any intellectual property issues in code.

#### ABOUT JBOSS ENTERPRISE MIDDLEWARE

JBoss Enterprise Middleware is an open source, standards based, comprehensive end-to-end enterprise middleware portfolio for Java application development, application and service hosting, content aggregation and presentation, data integration, and service integration and orchestration. JBoss Enterprise Middleware includes solutions across the entire application lifecycle: from innovative Web 2.0 development tools and frameworks, to SOA-based application and service deployments, to complete middleware management and monitoring. JBoss Enterprise Middleware is available via subscriptions that include certified software, industry-leading support, updates and patches, documentation, and multi-year maintenance policies. Learn more at [www.jboss.com](http://www.jboss.com).

#### ABOUT RED HAT

Red Hat, the world's leading open source solutions provider, is headquartered in Raleigh, NC with over 50 satellite offices spanning the globe. CIOs have ranked Red Hat first for value in Enterprise Software for four consecutive years in the CIO Insight Magazine Vendor Value study. Red Hat provides high-quality, affordable technology with its operating system platform, Red Hat Enterprise Linux, together with applications, management and Service-Oriented Architecture (SOA) solutions, including JBoss Enterprise Middleware. Red Hat also offers support, training, and consulting services to its customers worldwide. Learn more at [www.redhat.com](http://www.redhat.com).

Many of your peers and competitors already trust JBoss Enterprise Middleware for their mission critical applications and services. See who trusts JBoss.

[www.jboss.com/customers](http://www.jboss.com/customers)

#### JBOSS SALES AND INQUIRIES NORTH AMERICA

1-888-REDHAT1  
[www.jboss.com](http://www.jboss.com)