Evaluating Whether Open Source Application Server Software is Right for Your Company

Redguides

for Business Leaders

Ron Staerker

- Review of current open source software application servers
- Benefits and potential challenges with open source software solution
- Roadmap and recommendations for integrating open source software

Archive

Redbooks

Archived

Executive overview

This IBM® Redguide™ publication provides a framework for evaluating whether open source application server software is appropriate for your company. It begins with an overview of the philosophy and methodology inherent in open source software development, and highlights the potential benefits and pitfalls of incorporating this model in your organization. The IBM WebSphere® Application Server offerings, which range from an open source version to top of the line commercial solutions, are described and compared. Finally, some guidelines are provided to help you match your business requirements to the most appropriate and valuable technology.

Businesses today are under increasing pressure to reduce costs. Within the IT arena this desire to save money can translate into the use of open source software (OSS) among other strategies. The concept of free or very low cost software is certainly appealing, and open source software offers additional benefits as well, particularly from the point of view of the software developer community.

While open source has become an integral part of today's IT infrastructures, especially in the case of application server software, the decision to implement an OSS solution is one that requires careful consideration. Open source products are not appropriate for every need, and application server software in particular is not a “one size fits all” commodity. Furthermore, the promise of low cost can be a false economy when issues like service and support are factored in. Open source products, which are easy to acquire, might be challenging and expensive to deploy; integration and upgrade activities can be difficult and time consuming depending on your environment and your specific business requirements.

These possible challenges notwithstanding, IBM is firmly committed to supporting and advancing the acceptance of open source software. In fact, IBM offers a freely available open source version of its industry-leading WebSphere Application Server family, dubbed “Community Edition (CE).” IBM backs this offering with a full menu of support services to provide the most cost-effective and responsive assistance, based on the customer’s unique needs.

WebSphere Application Server Community Edition and other OSS application servers like those from Tomcat, JBoss, and Glassfish can be cost-effective solutions in limited environments when cost and simplicity are key considerations – and security, scalability, and manageability are less critical.

However, for most environments, those consisting of multiple application servers in clusters, open source application servers can be significantly more costly than their closed source

© Copyright IBM Corp. 2010. All rights reserved.

counterparts. Along the path of increasing complexity, heightened security, and application criticality are business demands that can be perfectly matched to the broad range of commercial offerings in the WebSphere Application Server portfolio. And, the WebSphere Application Server Community Edition offers a clear growth path to a networked WebSphere environment, ensuring your original steps in open source continue to pay dividends for years to come as your business grows and matures.

Archived

Introduction to open source software

Open source software (OSS) is computer software for which the source code and certain other rights normally reserved for copyright holders are provided under a software license that meets the *Open Source Definition* or that is in the public domain. This permits users to use, change, and improve the software, and to redistribute it in modified or unmodified forms. It is very often developed in a public, collaborative manner. The term “open source software” originated as part of a marketing campaign for free software.[1]

Open source is based on three concepts:

- A *development methodology* that defines a community approach to developing software, meritocracy of developers, and quality based on peer review.
- A *licensing approach* that provides free access to source code, conforms to one or more “Open Source Initiative” licenses, and prioritizes the rights of users and committers.
- A *community* of users and developers with open participation.

A good example of *development methodology* is defined by the Apache Software Foundation (ASF). The Apache Software Foundation provides a structured community approach for various OSS projects; many have become de-facto standards in the industry. These include projects like the Apache HTTP Server, Tomcat servlet and JavaServer Pages container, and Web Services. New projects can be submitted to the ASF for consideration. The project lives or dies based on community interest, building a developer base, earning the right to join the project community and access source code through individual contributions or *merit* (that is, meritocracy). An interesting point is that OSS allows downloads of source from the repository but updates to source are controlled by the project development community. Obviously the source updates must to be tightly controlled to ensure that concepts outside the scope of the project are not inserted and that peer review is provided within the community. OSS proponents should be aware that any changes that their developers make to open source does not guarantee that these changes will be available with the next OSS release.

A good example of *licensing approach* is also defined by the Apache Software Foundation Apache License, which defines the Terms and Conditions for use, reproduction, and distribution of the various ASF OSS projects. There are various OSS license types; the GNU General Public License (GPL), GNU Lesser GPL (LGPL), Berkeley Software Distribution (BSD), and the Massachusetts Institute of Technology (MIT) licenses, to name just a few. The open source licensing approach allows the authors of OSS to fine-tune these access and distribution privileges. Some licenses, like the Apache License, are better for commercial distribution because they release the user from sharing code with the author. Some licenses, like GPL, are better for the author because they require that changes or additions to the software be given back to the community (that is, Copyleft). It is sometimes difficult to understand the legal implications between these licensing approaches. Organizations that use OSS should understand where their developers are downloading software from and what OSS licensing agreements apply because there could be legal ramifications to incorrect OSS usage and embedding OSS code into commercial software.

A good example of the *community* of users and developers is supported by the policies and procedures of Apache Software Foundation projects. Users and developers are encouraged to submit changes, additions, and fixes to code to the project community. The community reviews code changes through peer review, and updates the code based upon the development methodology. Meritocracy allows developers to join the project community. The project community also determines, through discussion and voting, the future direction of the

[1] http://en.wikipedia.org/wiki/Open_source_software

project. Anyone can use the OSS binaries or download source for review provided that the user abides by the licensing approach.

Methodologies integral to developing open source software

There are several methodologies that are an integral part of OSS development. OSS proponents should understand these methodologies and how they might apply to their user experience.

- *Release early, release often (sometimes abbreviated RERO)*.

  This is a software development philosophy that emphasizes the importance of early and frequent releases in creating a tight feedback loop between developers and testers or users.[2] Many OSS projects release new production versions every few months. A new major release (for example, 1.0 or 2.0) might have new versions every few weeks or less due to bug fixes.

- *Versions are complete source builds*.

  The binary releases associated with an OSS project contain the entire application built from the community source repository. Patches to previous versions are rarely (if ever) provided. The advantage is that the user gets a complete integrated version. The disadvantage is it requires a reinstall of configurations and applications and a full retest of development, test, and production environments.

- *There are typically two versions released for OSS*.

  These versions include the last “stable” version release, which might contain minor bugs when released and is planned for production use. There might also be “alpha,” “beta,” or “milestone” version releases that are missing functions and contain bugs and are not for production. These alpha, beta, or milestone version releases might be released very often, possibly every day; they also get rolled into the next stable release. Unfortunately, a bug fix that is needed for your application might not be rolled into a stable version for quite some time.

- *OSS releases typically do not go through extensive Quality Assurance testing*.

  The user community provides testing of stable versions, and can also test alpha, beta, or milestone version releases along with the developer community.

- *“Free” software has less influence from users*.

  Is there a business requirement or nice feature that you want implemented in the OSS project? You can submit an enhancement request or even write the code, but there is no guarantee that the project community will implement your request.

- *OSS releases typically are not tested on a variety of hardware, OS, and JVM versions*.

  Java™ is used on many platforms – from PDAs to mainframes. But experience has shown that application servers are affected by the combination of hardware, OS, partitioning (LPARs), JVMs (for example, IBM versus Sun versus others), 32-bit versus 64 bit, and connections to resources like databases, caching, and security services. It might be best to stay with “generic” platform components for OSS, or you could be the first user to “certify” against your environment.

- *OSS does not come with a guarantee*.

  Hundreds of OSS-based projects have died for lack of support. Others have been replaced by comparable projects with better community support. Even within an OSS project, older versions become unsupported or might have a short life cycle.

[2] http://en.wikipedia.org/wiki/Release_early,_release_often

Potential benefits and challenges when considering open source software

Open source potentially offers both benefits and challenges. Users need to consider both before making a decision to implement.

Potential benefits

Users identify many potential benefits for using open source. Some of the major ones are:

- The cost of acquiring the software can be low or even zero. Furthermore, while support is usually charged for, users can normally decide whether to pay for support. Typically they would do this for mission critical deployments, but might not in the early evaluation or development phases.
- The software is easy to get hold of, either by downloading from the web, or often as part of a Linux® distribution. This frictionless access makes it easy for developers to try out the software for themselves, without the need to get purchasing involved. It also encourages the rapid prototyping and “do-it-yourself” computing environment popularized by scripting languages and the LAMP stack.
- Because the software is open source, it can often be obtained from multiple suppliers and often runs on multiple hardware architectures. Combined with the option of buying support, again often from multiple vendors, this all adds up to choice and flexibility for users.
- The peer review process of open source development has tended to produce higher quality code and enable the inclusion of innovations faster than is the case for normal proprietary software.
- Finally, the community approach allows developers to focus on value add components. In addition, open source is not just for developers: it also offers opportunities for users to get involved in interactions with the developers.

Potential challenges

There are a number of potential challenges that users should consider when looking at open source software.

- Is support available to the quality and service level agreements demanded by mission-critical use? Can the support vendor be depended on?
- Who integrates the various open source software components together, and who integrates open source with the existing IT infrastructure software? The customer? An integrating consultancy? A big IT vendor? Open source is a collection of communities, some of whom talk with each other and ensure integration, and some of whom do not.
- Are ISV applications available that build on the open source platform, and are they tested and supported to work on it?
- How mature is the open source offering? Does it have the functionality and scalability of competitive commercial offerings? Is there broad community and industry support for this offering? How are the developers of the open source offering going to make money? In other words, is there a sustainable business model behind it, and if not, how long is the offering going to last?
- How do you choose which open source software offering to use? Sometimes there are too many choices; it might be too early to pick the winner.
- Rapid prototyping and do-it-yourself (DIY) computing can be very effective at developing applications fast, but how maintainable are those applications, and how robust are they when they start to be changed?

Archived

- Application deployment for each OSS and commercial application server is different. There is no Java EE standard for deployment of applications and resources, so each application server does it differently. An application created to run on one application server will need to be migrated to another application server, so choose wisely. Note that application development using OSGi technology can help to resolve deployment incompatibilities.
- The Java Classloader for each OSS and commercial application server can be different. Tomcat uses a client classloader, meaning load from the application first, then from Tomcat. Most application servers like Geronimo, WebSphere, and GlassFish use a Parent classloader structure, which entails loading from the application server first, then the application. JBoss has its own way of loading classes, a mechanism called the Unified Class Loader. Thus an application created to run on one application server might not work on another application server without changes, so again choose wisely.
- Upgrades to the application server environment can be difficult and time consuming. There can be many releases since the time when the previous version was installed. Because OSS does not typically provide patches, a complete reinstall is required.
- OSS projects do not notify the user community of the status of issues. Someone needs to closely monitor all OSS projects for service issues such as security and defects to determine if a new version is required. Updates might cause problems with other integrated services.
- There might be migration issues requiring programming changes to implement newer versions of OSS. For example, there are migration steps to move from Tomcat 5.5 to 6.0. Each component or service added to the application (Web Services, JMS, Security, logging, and so forth) might also require migration and integration changes.
- Since OSS typically works on the Release Early, Release Often methodology, you will typically see new versions more often then commercial software. An organization might decide to support older OSS versions rather than updating, integrating, and testing production installs.
- Problem resolution on older OSS versions can be difficult. The community performs little testing on defects and problem resolution with older releases; these problems come from the user community. Committers are working on later releases. Due to the community-driven support approach, the older the version, the fewer people who are interested and able to support that release.
- Administration and development maintenance of older versions require a developer with an understanding of these versions and standards. Developers want to work on the latest and greatest development environments, not old standards. Developer job turnover can seriously affect this issue. Documentation on OSS is often not detailed, or is confusing between versions.
- The older version of the OSS application might include many components and services (Web Services, JMS, Security, logging, and so forth) that are also older versions. Problem resolution on older components can also be difficult. This can present a serious challenge when updating to the latest version, integrating, testing, and providing production support.

WebSphere Application Servers (Community and commercial versions)

This section describes the open source-based WebSphere Application Server Community Edition and compares it to WebSphere Application Server. The relationships to various open source projects are also described.

WebSphere Application Server Community Edition

The Apache Software Foundation *Geronimo* project was formed in August, 2003 with the objective to develop an Apache Licensed version of a J2EE 1.4 application server. It benefits from a diverse set of world-wide developers contributing to the project. Geronimo is a free download and does not have any fees associated with its use, commercially or otherwise.

The Geronimo mission was to create a Java EE certified application server that is offered under the Apache Software License, one of the most liberal open source licenses. It was not the objective of Geronimo to write a Java EE stack from scratch, but rather to leverage the best in breed open source projects that are available and allow them to become a part of the Java EE assembly. Geronimo collaborates with and uses successful middleware projects such as Tomcat, OpenEJB, ActiveMQ, Axis, OpenJPA, and many other popular open source offerings.

IBM has its own prepackaged version of Geronimo called *WebSphere Application Server Community Edition*, which is free to download and use, but which is backed by IBM support in case you need it. IBM also sells support for the Apache Geronimo product, employs several key Geronimo committers, and contributes code to the project.

It is easy to get started with WebSphere Application Server CE. The download is only around 80 MB, so developers can quickly download, install, and get going with it, usually in less than 10 minutes.

WebSphere Application Server CE starts with the Apache Tomcat, which developers are already familiar with. It then adds the Geronimo kernel, which provides a framework for adding other services like the custom business services your developers might have had to integrate if they were using just Tomcat. WebSphere Application Server Community Edition further bundles the leading components that developers need, like web services, security, and messaging. All these services come “in the box,” already integrated and tested with WebSphere Application Server Community Edition, so your developers do not have to worry about any custom integration work, version compatibility issues, or licensing issues.

WebSphere Application Server CE also supports creating custom server assemblies that contain only the functional components required by your applications. For example, a custom assembly containing the WebSphere Application Server CE kernel, Tomcat, and Axis could be created for a web application that includes web services. The WebSphere Application Server CE server size, memory requirements, and startup time would be greatly reduced with this custom assembly. Custom assemblies can be created using the console or by command line.

WebSphere Application Server CE includes a console that allows the user to configure and maintain WebSphere Application Server CE through a graphical user interface. This easy-to-use, browser-based web console provides functions for administrators and developers. The console supports managing the application server, including deploying and removing applications, configuring system services, and tuning the application server’s

Archived

performance. This means that you can easily tune WebSphere Application Server CE via the console, and your changes take effect immediately, without server restart.

Configuration of many WebSphere Application Server CE components is done through menu wizards that take the user through a short interview and provide a full plan configuration that is immediately available. Many of the wizards not only allow the configuration to be immediately deployed in WebSphere Application Server CE, but also offer an XML representation of the configuration that can be copied and pasted into an application plan file. WebSphere Application Server CE offers these wizards for database pools, security realms, web connectors, JMS topics and queues, JMS connectors, and keystore management.

Some of the key components in WebSphere Application Server CE are:

- The EJB engine component makes use of the Apache OpenEJB project. This lightweight EJB server is fully EJB 3.0 specification compliant and also supports earlier versions (2.1, 2.0, and 1.1).
- TranQL is the connector framework for WebSphere Application Server CE. TranQL is a connector framework made for building persistence engines. The connectors are bundled as a Resource Adapter Archive (RAR) file and deployed as any other packaged application in WebSphere Application Server CE. Users can configure connectors as server-wide, specific to an application, or specific to a single module.
- WebSphere Application Server CE uses Apache ActiveMQ as its messaging component. ActiveMQ is a lightweight messaging container that supports JMS v1.1 and includes JCA 1.5 resource adapters.
- Support for the Java Persistence API developed in the OpenJPA open source project. This API simplifies the development of Java applications using data persistence and is comparable to the JBoss Hibernate project.
- Support for new application standards, such as Servlet 2.5, JSP 2.1, JTA 1.1, and JavaBeans Activation Framework (JAF) 1.1, which simplifies development and improves transaction support and security.
- For security, WebSphere Application Server CE fully implements Java Authentication and Authorization Service (JAAS) and Java Authorization Contract for Containers (JACC - JSR 115). The security architecture is based on the pluggable GBean framework and the notion of Login Domains and Security Realms. This enables the building of sophisticated policies for application security. Security configuration can be done through the use of WebSphere Application Server CE plan files or through wizards in the WebSphere Application Server CE web console.
- WebSphere Application Server CE comes configured to use Apache Axis as its web services component and implements web services as required by the J2EE 1.4 and Java EE5 specification.
- WebSphere Application Server CE supports clustering of the web tier with the Tomcat web container.
- Deployment fully supports the JSR-88 specification and offers the ability to deploy applications using command line tools, using the web administration console, and using a hot-deploy directory.
- WebSphere Application Server CE supports database pool resources. The console includes a wizard for creating and testing database pool resources. WebSphere Application Server CE supports the use of free databases such as Apache Derby, IBM DB2® Express-C and MySQL, as well as the most popular commercial databases such as IBM DB2, MS SQL Server, and Oracle.

WebSphere Application Server CE works with various Application Server Tool Kits, Eclipse, and Rational® development tools, minimizing your learning requirements. WebSphere

Application Server CE offers an Eclipse WTP adapter that supports various Eclipse versions. The WTP adapter is installed in Eclipse and provides a tool for deploying and testing Java EE assets to a WebSphere Application Server CE server. In addition, WebSphere Application Server CE is compatible with popular open source frameworks and technologies such as Spring, Hibernate, Struts, and AJAX, which allows your team to simplify and accelerate development of dynamic applications and web front ends.

Finally, you can get support for all this from IBM so that you don't have to track down and solve problems on your own. This means you can write applications that you can rely on, and deploy them with confidence to WebSphere Application Server Community Edition. Your developers can concentrate on writing your core business applications. IBM can provide three levels of support assistance for WebSphere Application Server Community Edition:

- Entry
- Enhanced
- Elite

The levels vary by support hours, response time, and number of technical contacts.

An important feature of WebSphere Application Server CE support is Developer Assistance. With open source products, the developer has to look through forums and chats to resolve a question, with no guarantee that someone will respond. With Developer Assistance, developers can call or use the internet to post a PMR (Problem Management Record) to IBM support with their questions on issues like design, deployment, and tuning, and an IBM developer will help answer the questions.

It doesn't take a significant amount of developer time creating, maintaining, and debugging an OSS application server without support to justify WebSphere Application Server Community Edition support costs. One Saturday night call from your customer might justify the expense. This also applies to the rest of the WebSphere family. Migration resources and other information are available at the WebSphere Application Server Community Edition Developer's website (http://www.ibm.com/developerworks/spaces/wasce).

There is a clear growth path within the WebSphere family to handle thousands of users. WebSphere Application Server CE documentation includes migration guides that assist in the effort to migrate applications to WebSphere Application Server CE, as well as in migrating applications from WebSphere Application Server CE to WebSphere Application Server.

Tomcat application deployment and migration

Instead of deploying applications on Tomcat from the Apache Software Foundation, WebSphere Application Server Community Edition can provide integration and support services for Tomcat.

Tomcat applications can migrate to WebSphere Application Server CE because the Tomcat Servlet container version 6.0 is part of WebSphere Application Server CE. Besides supporting servlets and JSPs, WebSphere Application Server CE further bundles the leading components that developers need, like web services, security, logging, and messaging. All these services come “in the box”; they are already integrated and tested with WebSphere Application Server CE, so developers don't have to worry about any custom integration work or version compatibility issues. In addition, the WebSphere Application Server CE Administrator console is a powerful tool that allows developers to define Tomcat connections (HTTP, HTTPS AJP), database pools, and security realms without any coding, configuring of XML files, and without restarting the server. Like Tomcat, WebSphere Application Server CE can be used for commercial off-the-shelf or turnkey software packages where the servlet

container and HTTP listener is embedded but also supports other components within a web application. Therefore developers don't create a maintenance burden for themselves and administrators that detracts from their ability to address their business problems and build application projects quickly that deliver value to the business using a standard deployment container.

From the JavaWorld .com article “Is Tomcat an Application Server?”[3]

> *“It is not uncommon for Java enterprise applications to scale along the lines suggested by the example in this article, growing in complexity as new requirements are added. While Tomcat might suit your initial application requirements, it could become problematic in the long run, as the system gradually requires more complex deployment, management, and monitoring solutions. A Java EE application server (such as WAS CE or WebSphere) is more scalable than a Web server, providing tight integration between containers and deployment contexts for each additional technology. In many cases, choosing a Java EE app server is the more cost-effective solution for the long run.”*

Comparing the IBM WebSphere Application Server family to open source solutions

As the foundation of the WebSphere software platform, WebSphere Application Server is the industry’s premier Java-based application platform. For maximum flexibility, the WebSphere Application Server family offers a variety of configuration options and support for multiple business models and deployment platforms, including the open source-based option, WebSphere Application Server Community Edition. So whatever your needs, or however they change, there is a solution to suit you.

The IBM WebSphere Application Server family achieves its advantage by delivering an effective and efficient application foundation that helps you meet business objectives at the lowest overall cost. The following concepts clarify the WebSphere advantage:

- Use “right fit” deployment; not the concept that one OSS application server will fit all.
- While some OSS application servers might tempt customers with free or low cost license fees, ensure that the Total Cost of Ownership is considered.
- WebSphere helps you drive customer satisfaction with highly effective performance, offering faster transactions, less downtime, greater scalability, and streamlined security.
- For mission-critical environments, WebSphere delivers advanced, high availability features so you will experience less down time and your customers will have continuous access to their applications.
- WebSphere helps you reach customers in new ways by enabling application innovation that gives you more options in reusing existing applications as well as the ability to deploy the latest technologies with Feature Packs without wholesale upgrades.
- WebSphere helps reduce costs through efficient development and management, integrated and centralized tools, and up-to-date and thorough documentation.

These concepts are explained in greater detail in the following sections.

[3] http://www.javaworld.com/javaworld/jw-01-2008/jw-01-tomcat6.html

“Right fit” deployment: One OSS application server does not fit all

Customers seek a range of infrastructure to address the varying needs of small to enterprise scale applications. IBM WebSphere delivers the broadest range of offerings, from dynamic scripting with WebSphere sMash, to OSS with WebSphere Application Server Community Edition, to efficient and affordable delivery and management of business applications with WebSphere Application Server and WebSphere Application Server - Express, to higher performance with WebSphere Application Server ND, WebSphere Application Server for z/OS® and WebSphere Virtual Enterprise. The following list provides an overview of WebSphere Application Server offerings:

- WebSphere Application Server

  Efficient delivery and management of business applications and services of all types while reducing application infrastructure costs
- WebSphere Application Server - Express

  An affordable, ready-to-go solution to build dynamic web sites and applications
- WebSphere Application Server Community Edition

  Lightweight Java EE 5 application server based on open source Apache Geronimo
- WebSphere Application Server Network Deployment

  Delivers near-continuous availability, with advanced performance and management capabilities, for mission-critical applications
- WebSphere Application Server for z/OS

  Takes advantage of the qualities of service of IBM z/OS
- WebSphere Application Server Hypervisor Edition

  Optimized to instantly run in VMware and other server virtualization environments
- WebSphere Virtual Enterprise

  Helps you consolidate application servers and maximize utilization while monitoring application health
- WebSphere Cloudburst Appliance

  Speeds application deployment to cloud and virtualization environments
- WebSphere Application Server for Developers

  Efficient development and innovative features of WebSphere Application Server V7 at no charge
- WebSphere sMash

  A development and execution platform for quickly building agile, web-based applications

WebSphere Application Server Community Edition and WebSphere sMash have been available at no cost to developers since their introduction. IBM now provides WebSphere Application Server for Developers at no cost as well, helping developers to quickly get started on their projects and removing barriers to procuring software that developers often face within their own companies.

WebSphere CloudBurst™ will allow developers to more easily manage their virtual instances of WebSphere Application Server and set up flexible development environments.

IBM participation in open source initiatives

IBM is a strong supporter of open source software, as evidenced by the company’s endorsement and advancement of the Linux operating system. IBM participates in and

Archived

contributes to more than 150 open source projects. Examples of IBM open source participation are:

- Eclipse - IBM contributed Eclipse to the open source community.
- Apache Derby - This is an open sourcing of the IBM Cloudscape database.
- Linux - IBM regularly contributes code to Linux (including improvements in scalability, networking, serviceability, performance, availability, standards, security, and file systems).
- IBM maintains and contributes to several Apache projects, among which are WSIF, WSIL, Xerces, Xalan, Apache HTTP server, Axis, Derby, Geronimo, and Tuscany.
- Apache Aries - This enables an enterprise OSGi application programming model and is available with the WebSphere Application Server V7 Feature Pack for OSGi Applications.
- DeveloperWorks - This is the IBM web home for many open source projects that cover Java, web services, autonomic computing, XML, wireless, security, grid, and so forth.

Consider the total cost of ownership

Succeeding in today's smarter world not only means utilizing technology to create intelligent, instrumented, and interconnected solutions, but also leveraging that technology to minimize costs and do more with less. However, determining the lowest cost solution isn't simply a matter of comparing two vendors' prices, but assessing how the technology and features provided by those solutions can influence their total cost of ownership. The right solution will not only have the lowest cost, it will keep your applications running smoothly and help you meet your IT objectives.

With ever-increasing pressure to reduce costs, many companies are specifically looking to open source as a way to save money, which might mean considering an open source product like Tomcat, JBoss, or Glassfish. Open source application servers can certainly be cost-effective solutions in single server environments running one or two web applications. In these cases cost and simplicity are key considerations and security, scalability, and manageability are less critical. For these scenarios IBM offers IBM WebSphere Community Edition, an open source-based application server based on Apache Geronimo with no license costs. WebSphere Application Server and WebSphere Application Server - Express can also be used, with the further advantage of compatibility with WebSphere Application Server ND environments.

However, for environments consisting of multiple application servers in clusters (which constitutes most environments), open source application servers can be significantly more costly than their closed source counterparts. These environments, which often support business-critical, company-wide Java-based applications, can benefit from the highest qualities of service with respect to security, scalability, and manageability, qualities that are delivered through WebSphere Application Server Network Deployment. In contrast, using an open source application server in clustered environments can mean needing to hire additional, higher skilled administrators, suffering from insufficient performance and scalability, struggling with insufficient documentation, dealing with cumbersome security, or not having access to the latest innovations and technologies, all of which can significantly increase total cost of ownership and negatively impact your business success.

While some OSS application servers might tempt customers with low cost or no license fees, an IBM Business Value Assessment showed that license costs only account for 14% of the total cost of application server ownership. The remaining 86% of costs consist of hardware, training, support, development, and management costs. While WebSphere Application Server might have a higher initial acquisition cost, over time it makes up the difference by having lower administration costs, better performance, better reliability, and less overall risk.

IBM WebSphere Application Server achieves its cost advantage by delivering an effective and efficient application foundation that helps you meet business objectives at the lowest overall cost. The IT Manager can understand that there is an IBM solution in WebSphere, and the cost of retraining or losing their WebSphere administrators, along with adding migration costs to the budget, might not justify the perceived cost savings in switching to an open source product like Tomcat, JBoss, Glassfish, or even WebSphere Application Server Community Edition.

Drive customer satisfaction with highly effective performance

For business-critical environments, an IBM WebSphere application foundation will help you deliver higher quality application performance, and an enhanced experience for customers that OSS can't match. This not only means that your applications will run fast, but they will have higher availability and scalability, and the most dependable security. When your customers demand high performance and reliable applications, IBM is the clear choice over open source products like Tomcat, JBoss, or Glassfish.

Along with other companies, IBM uses the SPECj benchmarks to compare their application server performance with that of others. In fact, according to the independent SPECjAppServer2004 benchmark, WebSphere Application Server ND provides the second highest performance result in the industry.[4] Only Glassfish has participated in SPECj benchmark testing. JBoss has never provided a SPECj benchmark testing.

IBM is the first vendor in the industry to publish performance results based on SPECJEnterprise2010, the latest benchmark for businesses developing modern applications that utilize Java EE 5.0 for streamlined development to dramatically reduce costs, improve performance, and speed time-to-value.[5]

IBM also provides superior scalability over open source products like Tomcat, JBoss, or Glassfish, allowing you to spend less on additional hardware and software. If you need multiple processors to handle your transactions, this means real cost savings for you, because for a given workload on WebSphere Application Server, you will need less hardware than you would to run the same workload on OSS. Furthermore, with the extra hardware that OSS servers can require will also come more hardware support costs, more datacenter space, more power consumption, more OS licenses and support costs, and more support requirements. Even at moderate workload volumes, these costs can be significant.

Drive customer satisfaction with high availability

For mission-critical environments, IBM delivers advanced, high availability features so you will experience minimal down time for your applications. These high availability features include advanced clustering, data replication services, and unique workload distribution that enables applications to handle more concurrent users than open source products like Tomcat, JBoss, or Glassfish.

Finally, IBM WebSphere Application Server helps you avoid costly security exposures by providing granular and flexible control over security through a simple to use management tool, helping you deliver more secure applications that are easily managed by production operations. In contrast, most OSS application servers use manual file configurations to set up security.

[4] http://www.spec.org/jAppServer2004/results/jAppServer2004.html
[5] http://www-03.ibm.com/press/us/en/pressrelease/29145.wss

When cost and simplicity are greater concerns than performance and availability, open source solutions, like WebSphere Application Server CE, can often meet your needs. In the majority of environments when performance and qualities of service are critical to your business, you should not compromise with open source products like Tomcat, JBoss, or Glassfish when you can depend on the capabilities of WebSphere Application Server ND.

Reach customers in new ways with application innovation

Not only does WebSphere deliver highly effective performance, but it provides you with the technology to deliver innovative new applications, helping you reach customers in new ways and drive revenue growth. WebSphere Application Server helps you achieve this innovation through faster access to new technologies, targeted feature-specific upgrades, and broad support for integrating existing software in new ways.

IBM is simplifying the way you use WebSphere Application Server with the release of "Feature Packs." To balance our customers' desire for less frequent releases, while still making the latest standards available to those customers who need them, IBM has introduced Feature Packs. With Feature Packs, customers can selectively take advantage of new standards and features while maintaining a more stable internal release cycle.

One key benefit of Feature Packs is that they provide you with greater access to new technologies, standards, and programming models. This means that you can use the latest standards and programming models to deliver cutting edge solutions to your customers today, rather than waiting for OSS application servers like JBoss to support them in their next release or attempting to integrate a new technology with your application using Tomcat.

Many of the Feature Packs are based on OSS projects. For example, OSGI is based on Apache Aries, Java Persistence is based on Apache OpenJPA, SCA is based on Apache Tuscany, and web services is based on Apache Axis.

Feature Packs enable you to perform isolated upgrades of only the specific features and new technologies that you want, such as SCA, CEA, Web 2.0, and OSGI applications – without performing a full upgrade of the application server. In addition, the IBM WebSphere Application Server Feature Pack for Dynamic Scripting enables application developers to support dynamic scripting programming model languages like PHP and Groovy and deploy on WebSphere Application Server. This helps maintain a more stable application release cycle and minimizes disruption of operations to add new capability and address new standards. Meanwhile, open source products like Tomcat, JBoss, and Glassfish typically require users to perform a full application server upgrade to add new features, creating a disruptive proliferation of application server versions and unnecessary barriers between your customers and the latest technologies.

Available feature packs

The following feature packs are available as of this writing:

- IBM WebSphere Application Server Feature Pack for OSGi Applications and Java Persistence API (JPA) 2.0
- IBM WebSphere Application Server Feature Pack for Dynamic Scripting
- IBM WebSphere Application Server Feature Pack for XML
- IBM WebSphere Application Server Feature Pack for Communications Enabled Applications
- IBM WebSphere Application Server V7 Feature Pack for Service Component Architecture

- IBM WebSphere Application Server Feature Pack for Web 2.0
- IBM WebSphere Application Server Version 6.1 Feature Pack for EJB 3.0
- IBM WebSphere Application Server Version 6.1 Feature Pack for Web Services

WebSphere Application Server enables innovation by providing support for the broadest range of existing packaged and legacy applications, thereby permitting new services to be created and existing applications extended to the Web to improve the user experience. Tomcat and JBoss have no certified adapters for Siebel, SAP, PeopleSoft, or CICS®, which can result in reduced productivity and the need to reinvent or recode existing assets. As a result, a WebSphere Application Server solution allows you to spend your time and money unlocking new value from your existing applications, instead of waiting for development to rewrite code to make them work with JBoss.

Consider WebSphere instead of Tomcat for application deployment. All versions of WebSphere support the Java Servlet and Java ServerPages (JSP) standards, just as Tomcat does. WebSphere further bundles the leading components that developers need, like web services, security, and messaging. In addition, the WebSphere Administrator console is a powerful tool for developers and administrators.

Reduce costs with efficient management

The WebSphere Application Server family infrastructure not only makes your applications more effective and innovative, it also saves you money by enabling your employees to work smarter. Many factors contribute to this difference, including the extensive features for administration of applications included with IBM WebSphere Application Server ND. In contrast, Tomcat does not include a supported centralized management tool and the JBoss AS 5 JON console has management capabilities that are immature and incomplete relative to IBM capabilities. WebSphere Application Server ND also provides LDAP, cache, and WLM servers at no additional charge, saving money and simplifying compatibility.

WebSphere Application Server ND eases your management burden with a comprehensive set of tools for common administrative tasks such as backups, restores, cluster management, and remote application deployment, making complicated administrative tasks far simpler to perform than is true with open source products like Tomcat, JBoss, or Glassfish.

Finally, IBM reduces risk for your business and your customers by providing accurate and well-written documentation that fully specifies compatibility requirements, thereby helping to lower staff training costs and avoid incompatibility. OSS such as Tomcat and JBoss AS 5 offers limited documentation and does not adequately specify compatibility requirements with other software, driving users to a time-consuming search of forums and wikis for evidence of implied compatibility. You can rely on clear documentation from IBM to avoid the costly downtime, tedious searches and unexpected incompatibilities that come from unnecessary guess work.

Summarizing the benefits of WebSphere

WebSphere's effective, efficient, and innovation-enabling application foundation provides the best technology and features to create real value for both you and your customers, while reducing costs by providing more efficient development and management. With WebSphere Application Server Network Deployment as the foundation of your business-critical application infrastructure you can satisfy your customers with highly effective performance, reach them in

Archived

new ways with the latest innovative applications, and reduce your costs with more efficient development and management. For less critical applications open source based WebSphere Application Server CE can meet your needs. Lower IT costs are achieved with superior systems, and IBM and the WebSphere Application Server family provide the agility, the technology, and the simplicity to deliver the best solution at the lowest cost.

Whether you are creating a business-critical Java EE solution for millions of concurrent users, a light weight scripting-based solution for rapid time-to-value, or need interoperability with .NET, WebSphere has the goods to meet your needs. WebSphere Application Server provides the SOA run time of choice and keeps your company at the forefront with Web 2.0 capabilities.

IBM offers customers the broadest product range of application infrastructure, from WebSphere Virtual Enterprise at the high end to WebSphere sMash at the low end. No other vendor can match the breadth of the IBM WebSphere Application Server portfolio.

Questions and considerations to explore OSS usage

This section presents key questions and considerations to help you determine the suitability of open source software solutions for your business. It also provides recommendations on how to integrate open source software into your existing IT applications.

Software experts and researchers on open source software have identified several advantages and disadvantages. Open source is not an end in itself, but it might provide real IT benefits, which in turn can provide real business benefits.

The simple question *"Should we use open source software?"* must really be posed at a more granular level.

To evaluate the appropriateness of OSS in your environment you must answer all of the following questions instead:

- Will using OSS improve our IT infrastructure?
- Will it make our environment more reliable?
- More secure?
- Faster?
- More flexible?
- Will OSS lower the cost of our IT function?

And these IT questions in turn must support the following business level questions:

- Can open source as part of an IT infrastructure help our business succeed?
- Can it enable us to solve our problems today, grow profitably for the future?
- Will it help us integrate better within and beyond our company, and enable us to respond quickly to opportunities and threats?
- Does a 5 year OSS direction justify changes to the current IT infrastructure (migration, development, support, administration)?

There are cases where open source has indeed achieved all this. By focusing first on the business and wider IT considerations, you can determine whether open source software might be able to help your business succeed as well.

Archived

Planning for open source software integration

The following recommendations suggest a basic approach for planning the integration of open source software in your environment:

1. Insist on open standards

   Insist on open standards both for open source and private source commercial software. This increases choice and flexibility.

2. Consider using a mixture of open source and commercial software.

   Consider both open source and commercial (private source) software. Don't build a separate strategy, but integrate open source into your existing strategy. When evaluating open source alternatives, consider how these products will integrate with existing technology in your IT environment and determine how you could migrate to more enterprise-strength offerings if your performance and scalability needs increase. In addition, take a broad view of TCO when evaluating open source and commercial offerings. In addition to up-front costs, consider support costs, administration and developer costs (including training), migration costs, complexity, and so forth. In general, up-front costs account for just a small percentage of the overall TCO.

3. Evaluate the maturity of open source.

   Evaluate the maturity of open source projects before committing to them, especially in the areas of community, sustainable business models, and ecosystems. Open source projects that are backed by a strong “open” community present greater value and lower risk than vendor-controlled communities. Open communities protect against vendor lock-in and shield customers from the risk that a community controlled by a single company goes out of business or gets acquired. Make sure the community is vibrant and active and that the supporting ecosystem of ISVs and corporate backers is strong. Also ensure that the company from which you are acquiring the open source product and support is financially viable in the long future. You want to make sure your open source vendor is around to support you 2, 5, 10 years from now.

4. Establish policies for working with open source software.

   Establish company-wide policies for working with open source software right from the start. Implement a management system to review and track all use of open source software, and use due diligence to review the licensing for every open source software package used. Open source has a way of finding its way into an enterprise in a stealth-like manner. For example, many departmental developers use open source to rapidly prototype without going through procurement. It is important for enterprises to understand where and how open source is being used and to understand how it relates to and impacts the corporate software standards that have been mandated. This knowledge will help your enterprise better leverage open source and prevent the potential disruption undetected use might cause.

5. Be pragmatic in your approach.

   Run a pilot or proof-of-concept at an early stage and make decisions based on business factors and technical considerations just as you would for commercial software. Start small by picking one area where you think open source might be beneficial. Consider the costs, along with integration, interoperability, and migration scenarios, and then run a pilot to establish benchmarks for cost savings and other business-related benefits. Once satisfied with the results of the pilot, plan on a staged implementation to reduce disruption to operations and ensure a smooth migration.

6. Consider the big picture.

   Will your OSS choice fit future requirements? Consider the future complexity of applications, concurrent user requirements, database requirements, and other factors that

will require your application server environment to grow. Today's choice might not fit requirements five years from now.

Summary

Open source software, in particular application server software, is rapidly gaining acceptance and market share – user share, actually – because of its ability to combine high quality and innovative features in a low cost or no cost solution.

Building an application infrastructure with OSS has certain advantages, but also presents a number of potential challenges. It is crucial to consider these issues, not just the potential cost savings, when evaluating whether open source or commercial software, or a combination of both, is the best strategy for your organization. The size and complexity of the business, the nature of the applications to be run, skill set of the users, security requirements, and so forth, must also be factored into an evaluation of the appropriateness of OSS.

The IBM WebSphere Application Server family of offerings covers the entire field – from simple, single server environments to global networks. This unparalleled variety of solutions run the gamut from a Community Edition with no licensing cost to enterprise-wide, global network deployments. At every level, the WebSphere portfolio delivers reliability, high performance, security, manageability, scalability, and cost-effectiveness.

Other resources for more information

- WebSphere Application Server:
  http://www.ibm.com/software/webservers/appserv/was/
- WebSphere Application Server Community Edition:
  http://www.ibm.com/software/webservers/appserv/community/
- WebSphere sMash: http://www.ibm.com/software/webservers/smash/

The author who wrote this guide

This guide was produced by a technical specialist working for the International Technical Support Organization (ITSO).

Ron Staerker is a technical specialist in the IBM Software Group, Application and Integration Middleware Software, working in Los Angeles, CA, USA. He has more than 38 years of experience in the computer industry, including positions as developer, consultant, and sales engineer to organizations ranging from small businesses to Fortune 500 companies in various industries. He has designed and developed applications using everything from machine code to Java, on hardware from microcomputers to mainframes. Since joining IBM, Ron has been a Technical Consultant for the World Wide WebSphere team working on development software and open source based products. He has authored and co-authored Redbooks® and other articles on Java development and on using WebSphere Community Edition.

Thanks to the following people for their contribution to this project

Adam Orentlicher - IBM Software Group, Application and Integration Middleware Software
WebSphere Offering Team Lead
Durham, NC, USA

Erik Kristiansen - IBM Software Group, Application and Integration Middleware Software Product Manager - WebSphere Application Server Feature Packs, WebSphere Application Server CE, WebSphere Application Server Express
Durham, NC, USA

Now you can become a published author, too!

Note: Performance information in this paper is based on measurements and projections using standard IBM benchmarks in a controlled environment. Actual performance in a user's environment might vary.

Here's an opportunity to spotlight your skills, grow your career, and become a published author - all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks
- Follow us on Twitter:

  http://twitter.com/ibmredbooks
- Look for us on LinkedIn:

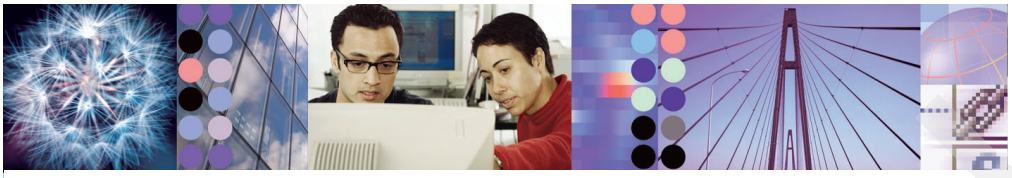
  http://www.linkedin.com/groups?home=&gid=2130806
- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm
- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

Archive

Contact an IBM Software Services Sales Specialist



IBM

Start SMALL, Start BIG, ... **JUST START**

architectural knowledge, skills, research and development . . .

that's IBM Software Services for WebSphere.

Our highly skilled consultants make it easy for you to design, build, test and deploy solutions, helping you build a smarter and more efficient business. **Our worldwide network of services specialists wants you to have it all!** Implementation, migration, architecture and design services: IBM Software Services has the right fit for you. We also deliver just-in-time, customized workshops and education tailored for your business needs. You have the knowledge, now reach out to the experts who can help you extend and realize the value.

For a WebSphere services solution that fits your needs, contact an IBM Software Services Sales Specialist:
ibm.com/developerworks/websphere/services/contacts.html

Archived

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

© Copyright IBM Corp. 2010. All rights reserved.

This document, REDP-4619-00, was created or updated on August 2, 2010.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at
http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

CICS®	Rational®	WebSphere®
CloudBurst™	Redbooks®	z/OS®
DB2®	Redguide™	
IBM®	Redbooks (logo) ®	

The following terms are trademarks of other companies:

Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Archived