WebSphere® software

Supporting open standards for Web services and the Java 2 Platform, Enterprise Edition (J2EE).

By Sam Caruso and Jeff Reser,
IBM Software Group

Contents

Introduction

This white paper discusses open standards for Web application development and deployment platforms. It describes the IBM WebSphere® software platform and positions current standards, such as Java™ 2 Platform, Enterprise Edition (J2EE) with emerging standards (around Web services) and other factors — scalability and quality of implementation — that affect a company's choice of Web application platform. The intended audience for this white paper is information technology (IT) professionals, developers, technologists or anyone seeking information on which to base a choice of a Web application platform. The content is technically oriented, with specific details about the standards supported by the WebSphere software platform and platform-based technologies.

The WebSphere software platform for e-business delivers the most powerful and flexible Web application server on the market, partly because of the broadest implementation of the widest range of leading-edge open standards. The application server is complemented by a range of products that leverage this foundation to provide functions, such as personalization and mobile computing. The WebSphere software platform includes an award-winning line of tools, including IBM WebSphere Studio and IBM VisualAge® for Java, providing a highly integrated development and deployment environment.

And now, the latest generation of tools, based on the WebSphere Studio Workbench platform, continue the award-winning functions and provide a common look and feel for the multiple development roles necessary to produce a Web-based application. Further, the WebSphere Studio Workbench platform is based on the open-source Eclipse technology and provides the means for third-party tool vendors to construct their next-generation tools in a way that provides a consistent look and feel and also tightly integrates other tools on the platform. For more information, visit www.eclipse.org and **ibm.com**/websphere/studio.

"WebSphere, with its Java-based J2EE technology framework, gives us tremendous business advantages."
— **Maynard Webb, President, eBay Technologies**

The WebSphere software platform delivers the fullest set of application program interfaces (APIs) for Extensible Markup Language (XML), enterprise Java and Web services across more than 35 platforms — including Microsoft® Windows NT®, Linux®, HP-UX and IBM S/390®. The WebSphere software platform delivers a robust application server that provides the J2EE technology-based platform integrated with tools that support Web services standards, such as Simple Object Access Protocol (SOAP) and Universal Description, Discovery and Integration (UDDI). By implementing these open standards in ways that preserve and enhance the server's scalability and reliability, WebSphere software platform delivers a highly productive environment for developing, deploying and integrating e-business applications. The incorporation of Web services opens a new door to building WebSphere software applications that dynamically interact with Web services using SOAP, UDDI, Web Services Description Language (WSDL) and XML.

IBM continues to contribute to the development of standards with active involvement in standards bodies, such as the World Wide Web Consortium, the Open Software Foundation, the Java Community Process Program and the Web Services Interoperability Organization (WS-I). This helps to ensure real customer needs are driving the improvement of existing specifications and introduction of new specifications. But this is only the first part of the process. After a standard is developed, it must be adopted and then delivered in a meaningful way in products. This white paper explains today's most important e-business standards and how they are implemented in the WebSphere software platform to deliver on the promise to make it easier for you to develop and deploy Web applications. And to do it all over again tomorrow.

"Whenever we have a complex, high-volume distributed application, we turn to the WebSphere software platform. It provides us with a broad set of services and tools to deliver a wide variety of solutions to any vertical market."
— **Steve Jones, Chief Marketing Officer, Trifolium, Inc.**

The value of open standards in theory and in practice

Open standards provide a way for those who are responsible for technology decisions to predict a certain level of functionality and interoperability from vendors they consider. Today's marketplace for Web application development technology demands open standards. If you develop applications using open standards-based technology, it helps ensure that the application you build is viable today and tomorrow. The open standards for Web applications are developing and expanding as you read this — and IBM is a major force behind them.

IBM is an innovator and leader in open standards for Web application development. IBM contributes the majority of the specifications and application programming interfaces (APIs) to the J2EE technology-based platform and expands the concept to other areas requiring standardization.

Evaluating open standards in an e-business platform

A vendor's support of open standards is an important element of consideration for you when choosing technology on which to base an application. The decision process behind choosing a technology, however, needs to go beyond looking for a simple statement of support to evaluating the implementation of the standards. When choosing a platform on which to base your Web application, you can use four questions to evaluate vendors and their Web application technologies:

1. Is the technology solid, today and tomorrow?
2. Does the technology address the whole picture?
3. Does the vendor include the platforms you invest in (or may invest in soon)?
4. Can the vendor help you apply the technology in the right way?

Is the technology solid, today and tomorrow?

First and foremost, build on a solid foundation. That means quality components. The J2EE specification is a good start. The next step is the quality of how that specification is implemented. When you use IBM technology, you are sure to get high-quality workmanship that implements the spirit of the law, as well as the letter of the specification. IBM technology is J2EE-certified, but the elements of the J2EE platform only specify what to implement, not how. The how may carry implications that can prove expensive or add complications later down the road. For instance, some competitors use relationships with third parties to provide large chunks of function. How solid are those third parties? How long will their relationships last? Are their development cycles in sync? Are their corporate objectives in sync today? How about tomorrow? IBM technology is rock solid, and it's provided by IBM.

Does the technology address the whole picture?

Next, make sure you see the whole picture. The standards for Web-based applications are still forming. Good standards exist today (for example, J2EE technology) but others are necessary to consider when you build your application. IBM has provided or contributed to more than 80 percent of the APIs and specifications that go into the J2EE platform, but J2EE technology is just the baseline of standards that need consideration.

"Not once have we had to go to IBM and say we have a bug; not in the 14 months we've been live."
— **Thomas Richards, VP, Nexstar Financial Group of St. Louis**

"Over time, there seems little doubt that IBM's enormous resources will enable it to dominate."
— **Eric Knorr, ZDNet**

IBM made the following contributions to the Java platform:

- *Java Transaction Service (JTS)*
- *Remote Method Invocation Internet Inter-ORB Protocol (RMI-IIOP), RMI running over IIOP Java Industry Award*
- *J2EE Connector Architecture (JCA), derived from IBM Common Connector Framework (CCF)*
- *National language support*
- *User Interface, 2D Graphics*
- *InfoBus, client publication and subscription service*
- *Security, Java Development Kit (JDK) 1.2 backport*
- *RealtimeJava, Java Industry Award*

IBM has also worked jointly with companies to develop other elements of the Java platform, including:

- *Enterprise JavaBeans (EJB) Specification*
- *JavaBeans, Component Assembly Specification*
- *Java Foundation Classes (JFC)*
- *Java Database Connectivity (JDBC)*
- *JavaSpeech APIs (JSAPIs)*
- *Java Multimedia Frameworks (JMF)*
- *Java Message Service (JMS)*
- *Java Management Extensions (JMX)*
- *Java Advanced Interface for Networking (JAIN)*

Today, the idea of reusable technology is paramount; build an element that provides a useful function and reuse it in as many places as possible. Extend that idea beyond your current application and you find yourself in the new exciting world of Web services. Imagine the benefits to production and quality when your application can access technology that is:

- *Written for a specific business purpose that your application requires*
- *Written by experts who specialize in that business purpose*
- *Accessible to you without the cost of development or maintenance*

The challenge is to find and access the Web service you need. IBM technology helps you with both. IBM is a major contributor to the UDDI initiative, a public registry for advertising Web services. UDDI provides the only existing forum for online registration of Web services. You can find the IBM implementation of this registry at **ibm.com**/services/uddi. It's another example of how IBM goes beyond the specification to deliver tools and implementations to help you unleash the value of the standard.

Choose IBM technology to take advantage of the Web services concept, providing:

- *Tools to help you access (or create) Web services elements*
- *Support for XML to exchange data*
- *Support for WSDL to communicate to a registry*
- *SOAP API to use Web services*

Integrating this technology with the application server rather than adding it on, or providing it through a third party, brings big benefits. If integrated with the application server, you can maintain the benefits of the application server and the Java programming model, namely:

- *Scalability. Workload balancing through a single-system image across clusters*
- *Security. Authorization control at the application or Web service level, as well as for methods within a service*
- *SOAP security. Basic authentication, asymmetric secure sockets layer (SSL), SOAP digital signature, symmetric HTTPS*
- *Transaction services. IBM DB2®, IBM CICS®, IBM IMS™*
- *Serviceability. Distributed debugger, trace*

IBM integrates all these technology elements with an application server that includes a Java API (UDDI4J) to make working with the UDDI registry easier. Web services and its related technology will be demanded by tomorrow's open standards; be prepared by building on the IBM foundation that provides access to these standards today. For more information about Web services on the WebSphere platform, visit **ibm.com**/websphere/wow/.

"We didn't have time for false starts. In fact, I can't imagine how the development time could have been shorter. We had very good support, however, from the IBM Austin Lab. That was one reason we chose IBM in the first place. We knew we needed a partner we could count on."

— David Guild, Vice President, Corporate Technology Services, Prudential Insurance

Does the vendor include the platforms you invest in (or may invest in soon)?

Another element of open standards implies portability. Select a vendor that provides technology on the platform you have today, and the one you may be using tomorrow. Today, there are a variety of platforms by different vendors. Hardware is different. Solutions exist for platforms from personal computers to mainframe systems. Some platforms exist in your environment today and you may be looking at emerging platform technology for tomorrow. IBM WebSphere software platform and platform-based technology supports all the major platforms today, including Linux, now emerging as a leading-edge deployment platform. Other supported platforms include Microsoft Windows NT, Windows® 2000, Sun Solaris operating system, IBM AIX® and IBM @server pSeries™ (formerly IBM RS/6000®), HP-UX, Red Hat Linux, TurboLinux, SuSE Linux, Novell NetWare, IBM OS/400®, IBM z/OS™ (formerly IBM OS/390®) and Linux for IBM @server zSeries™.

The WebSphere software platform is the only J2EE technology-based platform that leverages the capabilities of z/OS and IBM OS/390. Having an application server in close proximity to the enterprise data on OS/390 means faster transaction throughput. The internal communications of the OS/390 can transfer data from a database to EJB components and through the application server at speeds up to gigabits per second, providing the highest quality-of-service levels in the industry. What this means for you is choice and room to expand. With IBM and WebSphere software, you can develop your application on one platform with the confidence that it will run on the platform you may be using tomorrow.

Can the vendor help you apply the technology in the right way?

Consider also whether the vendor can answer your questions as you build your application. Can the vendor help you architect your application to ensure the most likely chance of success? Has the vendor done it before? Does the vendor offer publications to help you learn on your own? Can the vendor provide education, training and support for you when you need more help?

IBM has captured a wealth of information around best practices in building and running e-business applications. The IBM Software Strategy for e-business provides a guide and a best-practices view of developing technology for the Web today. See this at **ibm.com**/software/ebusiness. Next, visit redbooks.**ibm.com** and search on WebSphere software platform to see a list of no-charge publications that provide in-depth technical advice on a range of different e-business solutions. You can also visit the WebSphere software home page at **ibm.com**/websphere, and click the Education section to see a list of classes that can help you make the most of your open standards-based platform. And, there's a network of IBM consultants, IBM Business Partners and Web integrators standing by to provide services at your request. Use the IBM jStart program to help you get cutting-edge projects up and running fast! Read more about jStart at **ibm.com**/software/ebusiness/jstart.

Diving deeper

IBM WebSphere Application Server, Version 5.0 embodies the latest technology to support Web services in addition to certified compatibility with the J2EE 1.3 specification. The following sections explain in greater detail the standards for Web-based services and the J2EE-based platform, and how IBM WebSphere Application Server implements these standards.

Table 1. Web services-related technology included in WebSphere Application Server, Version 5.0

IBM WebSphere software platform-based technology	Description	Support
Simple Object Access Protocol (SOAP)	Industry standard for using XML to represent data and remote service calls for open access to Web services	SOAP, Version 1.1 gateway has been added. This includes integration of IBM SOAP for Java toolkit and parser runtime.
Web Services Description Language (WSDL)	Defines a service interface (like Interface Definition Language, or DNL) which specifies access protocols (like SOAP or SMTP)	WSDL, Version 1.1 processor and generator has been added.
Universal Description, Discovery and Integration (UDDI)	Registry interface definition for publishing and accessing Web services	UDDI, Version 1.0 interface has been added.
Extensible Markup Language (XML)	An extended tag-based language format for presentation and connectivity	XML parsing is supported.

Open standards for Web services

The WebSphere software platform integrates support for the latest industry standards that can be relied upon for the huge opportunity that exists for Web services. Three aspects of such implementations are:

- *Delivering new applications as Web services*
- *Converting existing applications to Web services*
- *Interacting with Web services from WebSphere applications*

The support for Web services in IBM WebSphere Application Server, Version 5.0 includes:

- *Integration of technologies from the Web services toolkit (from IBM alphaWorks®) directly with the WebSphere Application Server runtime component. The Web services toolkit continues to evolve on alphaWorks as well.*
- *Integration of IBM SOAP for Java toolkit and parser runtime (Apache.xml open source technology). This enables the capability of WebSphere software to receive, send, respond and create SOAP and WSDL messages. The transport on which the SOAP package is received is protocol independent. That is, it can be HTTP, JMS or an enterprise messaging environment like IBM WebSphere MQ (formerly IBM MQSeries®). Integration of the UDDI for Java (Java interface to UDDI registries) toolkit and parser runtime (also contributed by IBM to the open source community) enables you to register a WebSphere software application in a UDDI-compatible registry, and send, receive, create and respond to UDDI and WSDL messages. The transport on which these messages are received or sent is independent of the WebSphere Application Server function — that is, it could be HTTP, JMS, WebSphere MQ or another technology.*
- *XML support includes an improved XML parser runtime and an Extensible Stylesheet Language (XSL) processor.*

Tools support for Web services and XML development environment includes:

- *Web services tools for generating SOAP-based applications which are, in turn, utilizing WSDL and XML for communication with UDDI registries and other Web services*
- *Web services tools for extending existing applications into Web services*
- *Web services tools to enable connectivity from WebSphere applications to other Web services*

In addition, the next-generation IBM WebSphere Studio development environment is available for purchase or trial download at **ibm.com**/websphere/studio.

Deliver new applications as Web services

WebSphere software platform-based technology helps you to author, deploy and manage secure and authenticated Web services. Your services can be implemented as Java classes, Enterprise JavaBeans, JavaScript or IBM DB2 stored procedures. Your Web services can be secured using HTTP basic authentication, HTTPS and SSL encryption and digital signature. Support for digital signature follows the SOAP security specification recently submitted to the World Wide Web Consortium. A WSDL specification of a service interface written by your solution architects can generate a JavaBeans template. Your application development team can then fill in this template to implement the service. Optionally, you can publish your service description to a UDDI registry using the open source UDDI4J interface incorporated in the WebSphere Studio tool. You can package Web services as enterprise applications to deploy to the WebSphere Application Server for access over the Internet through SOAP over HTTP. The WebSphere Application Server administrator manages the security and life-cycle attributes of these services. Services that you implement as EJB components additionally enable management of transaction attributes and access control at the operations level.

Deliver existing applications as Web services

WebSphere technology also helps enable the transformation of existing applications into Web services. You can make all WebSphere software applications — and those accessible through Java technology-based connectivity — accessible over the Internet. These include WebSphere MQ messages, CICS transactions, DB2 queries and updates, IMS transactions, and SAP Remote Function Calls (RFCs) and business object application program interfaces (BAPIs).

Interact with Web services from your WebSphere software applications

You can incorporate any Web service made available by other vendors with your WebSphere software solution. The next generation of the IBM WebSphere Studio tool includes a UDDI interface so you can search registries for services providing the required function. Whether you obtain a service provider's WSDL description through UDDI, a link from a Web page or an e-mail attachment, the WebSphere Studio tool can use it to generate a Java proxy class that implements the interface. You can incorporate the generated Java proxy class with your application — thus, Web service interaction simplifies to Java object calls and responses.

Again, through WebSphere software support for integration with the leading application environments, you can easily interact with Web services from your new or existing applications. If the service includes a human user interface element, the WSDL reference can be added to JavaServer Pages (JSP) and delivered through a standard Web browser. Also, because the provider is offering its service through a Web standard interface, you can forget about the platform that hosts the service and the language that was used to develop it.

Open standards for the enterprise Java platform
Table 2 describes Java 2 Platform, Enterprise Edition (J2EE). J2EE, a Sun Microsystems, Inc. brand name, is the Java standard for enterprise application development and deployment. The platform specifications include the Java 2 APIs and technologies required to provide the minimum quality of service, compatibility, portability and integration defined by the J2EE platform.

In addition to these J2EE specifications, J2EE-based technology includes three other deliverables:

- *J2EE compatibility test suite* Verifies that a Java implementation complies with J2EE specifications.
- *J2EE reference implementation* Provides an operational definition of the J2EE platform and demonstrates its capabilities.
- *J2EE blueprints* Provide guidance on building enterprise applications that can run on a J2EE platform. J2EE blueprints focus on the development of JSP, Java servlets and EJB components that use resources, such as JDBC and JNDI, to run the applications. The wide range of application technologies and APIs shown in the following table can communicate with a variety of systems, servers and applications that can exist within a given network.

WebSphere Application Server and Java 2 standards
The following table shows how WebSphere Application Server compares with the full range of Java 2 APIs and related technologies. The following is a key to interpreting the table columns:

- *Java 2 technologies and APIs list the J2EE API or technology.*
- *WebSphere Application Server, Version 4.0 describes the support in the previous version of WebSphere Application Server.*
- *WebSphere Application Server, Version 5.0 describes the support in the latest J2EE 1.3-certified deliverable.*

Table 2. Java 2 platform summary

Technology	Description
Enterprise JavaBeans (EJB)	Server transactional components that are reusable and provide portability across application servers while implementing transaction services
JavaBeans Activation Framework (JAF)	Standard services to determine the type of an arbitrary piece of data and activate an appropriate bean component to manipulate the data
Java Interface Definition Language (JavaIDL)	Remote interfaces created to support Java-to-CORBA application communication. JavaIDL includes an IDL-to-Java compiler and an Object Request Broker (ORB) that supports Internet Inter-ORB Protocol (IIOP)
JavaMail	A protocol-independent framework to build mail and messaging applications; requires the JavaBeans Activation Framework API
J2EE Connector Architecture (JCA)	Used by tools vendors and system integrators to create resource adapters that support access to enterprise information systems that can be plugged into any J2EE product
Java DataBase Connectivity (JDBC)	The JDBC database access API that provides uniform access to relational databases, such as DB2, Informix, Oracle, Sybase, SQL Server and others
Java 2 Server Development Kit (SDK)	Provides the Java Virtual Machine (JVM) base with Java classes and basic routines required to execute Java applications
Java Message Service (JMS)	Supports asynchronous communication using either a reliable queuing or publish and subscribe programming model
Java Naming and Directory Interface (JNDI)	Provides access to naming and directory services, such as domain name server (DNS), Lightweight Directory Access Protocol (LDAP), Novell Directory Services and Common Object Request Broker Architecture (CORBA) COSNaming
JavaServer Pages (JSP)	A simple, fast and consistent way to extend Web server functionality and create dynamic Web content; JSP platform-based technology enables rapid development of Web applications that are server and platform independent
Java Transaction Service (JTS) Java Transaction API (JTA)	A distributed transaction management service and associated API based on the CORBA Object Transaction Service
Remote Method Invocation/Internet Inter-ORB Protocol (RMI-IIOP)	With RMI, creates remote interfaces for Java-to-Java application communication; the RMI-IIOP extension uses the CORBA-standard IIOP for communication
Java Authentication and Authorization Service (JAAS)	Provides a way for a J2EE application to authenticate and authorize a specific user or group of users to run it. This is a Java technology-based version of the standalone Pluggable Authentication Module (PAM) framework
Java Servlet	Server applications that execute within a Web application server to support dynamic HTML; the Java Servlet API gives Web developers a simple, consistent way to extend the functionality of a Web server
Extensible Markup Language (XML)	As it relates to the Java 2 platform, XML provides an extended format for deployment descriptors within Enterprise JavaBeans for the J2EE platform
Java API for XML Processing (JAXP)	Supports processing of XML documents by enabling applications to parse and transform these documents independently from a particular XML processing implementation

Table 3. Java 2 Platform, Enterprise Edition support in WebSphere Application Server

Java 2 platform-based technologies and APIs	WebSphere Application Server, Version 4.0	WebSphere Application Server, Version 5.0
J2EE Certification	Compatible, fully J2EE 1.2-certified and part of Sun Microsystems JCEE list	Compatible, fully J2EE 1.3-certified and part of Sun Microsystems JCEE list
EJB	Compatible, 1.1 including XML descriptor support for XML support within transactional applications	Compatible, 2.0, 1.1 including message-driven beans support
JDK	Compatible, JDK 1.3 across all supported OS platforms including Windows NT, Windows 2000, AIX, Sun Solaris, HP-UX, Linux, OS/400, and z/OS	Compatible, JDK 1.3.1 across all supported OS platforms
Servlet	Compatible, Servlet 2.2 API level	Compatible, Servlet 2.3 API level
J2EE Connector Architecture (JCA)	IBM Common Connector Framework supported	Compatible
JTS/JTA	Compatible, including distributed transactions — JTS for CICS, IMS integration and transaction management; JTA for relational database transaction management	Compatible, including distributed transactions
JMS	Compatible, WebSphere MQ native support, plus JMS-XA support for coordinated transactions across WebSphere MQ-enabled systems and relational databases	Compatible 1.02, with embedded JMS provider with point-to-point and publish/subscribe support and JMS-XA support
JDBC	Compatible across all databases, including DB2, Oracle, Sybase, Informix, and Microsoft SQL Server	Compatible across all databases, including DB2, Oracle, Sybase, Informix, and Microsoft SQL Server
JNDI	Compatible, JNDI 1.2 for EJB lookup and COSNaming	Compatible, JNDI 1.2 for EJB looking and COSNaming
RMI-IIOP	Compatible, fully supported for EJB-EJB and application server communications	Compatible, fully supported
JavaMail/JAF	Compatible, supports third-party vendor implementations for SMTP and MIME and e-mail (can utilize Lotus® Domino™ support for JavaMail as well)	Compatible, supports third-party vendor implementations
JAAS, JSSE (Java Secure Sockets Extension), JCE (Java Cryptography Extension) authentication model and secure sockets	Compatible, except for JCE	Compatible, implementing the JAAS pluggable
JAXP (Java API for XML Programming), XML, XML DOM/SAX (Document Object Model / Simple API for XML), XSL (eXtensible Stylesheet Language)	Compatible, IBM continues as industry leader with additional support for XML within EJB and XML within Web Services	Compatible, including enhanced support for JAXP
JavaIDL/CORBA	Compatible	Compatible, including support for CORBA 2.3 ORB
LDAP (Lightweight Directory Access Protocol)	Compatible, LDAP client/server support, support for Lotus Domino, Netscape, Tivoli® Directory, ActiveX Directory	Compatible, LDAP client/server
HTTP	Compatible, across multiple Web servers	Compatible, across multiple Web servers
Java RMI-IIOP	Compatible	Compatible

EJB

WebSphere Application Server, Version 5.0 fully supports Enterprise JavaBeans, Version 2.0, including full support for all three bean types (session beans, entity beans and message-driven beans), along with container-managed persistence. An enterprise bean is a body of code with fields and methods to implement modules of business logic. EJB, Version 2.0 adds the definition of message-driven beans that rely upon JMS as the messaging transport for managing asynchronous transactions.

JDK

WebSphere Application Server fully supports IBM Java Development Kit 1.3 across Windows NT, Windows 2000, AIX, Sun Solaris operating system, HP-UX, Linux, Linux for z/OS, OS/400 and z/OS. IBM implements JDK, Version 1.3.1 (part of the J2EE, 1.3 platform) to provide continued support of the J2EE standard and leadership in providing high-quality technology on the latest standards.

Java Servlet

The latest stable Java Servlet API (2.3) is supported. A servlet class extends the capabilities of servers that host applications accessed by way of a request-response programming model.

JavaServer Pages

WebSphere Application Server fully supports JSP, 1.2 with custom tags. As with JDK and servlet support, the goal of IBM is to deliver real business value to developers and customers through productive development environments like WebSphere Studio and with stable runtimes like WebSphere Application Server.

JCA

WebSphere Application Server fully supports and implements the J2EE Connector Architecture. JCA can be used by J2EE tools vendors and system integrators to create resource adapters, which in turn support access to enterprise information systems that can be plugged into any J2EE product. A resource adapter is a software component that allows J2EE application components to access and interact with the underlying resource manager. Because a resource adapter is specific to its resource manager, there is typically a different resource adapter for each type of database or enterprise information system.

JTS/JTA

WebSphere Application Server supports Java Transaction Services 1.0 and Java Transaction API 1.0. As part of the reference implementation for the J2EE platform, IBM has implemented a transaction manager that supports JTS/JTA. JTS/JTA lets application servers, built on the J2EE platform, take the burden of transaction management off the component developer. Developers can define the transactional properties of EJB-based components during design or deployment, using declarative statements in the deployment descriptor. The application server takes over the transaction management responsibilities. JTA specifies standard Java interfaces between a transaction manager and the parties involved in a distributed transaction system: the resource manager, the application server and the transactional applications. A JTS transaction manager component provides transaction services to the parties involved in distributed transactions: the application server, the resource manager, the standalone transactional application and the communication resource manager (CRM). WebSphere supports distributed transactions with EJB-based technology, where EJB components running in WebSphere Application Server can participate within the same transaction and utilize two-phase commit.

JMS

JMS simplifies development by allowing loosely coupled, reliable asynchronous interactions among J2EE-based components and legacy systems capable of messaging. WebSphere Application Server, Version 5 offers an integrated JMS subsystem based on market-leading WebSphere MQ technology. Customers also have the option to utilize the separately available WebSphere MQ environment, which is fully JMS-compatible and offers a superset of the functionality in the JMS 1.0.1 specification.

JDBC

WebSphere Application Server includes full Java Database Connectivity 2.0 support for a wide range of JDBC 2.0-compatible relational database systems, both as container-managed persistence databases and with distributed, heterogeneous transactions. The list of databases includes DB2, Informix, Microsoft SQL Server, Oracle, Sybase, Versant and others. WebSphere Application Server supports distributed database transactions across heterogeneous DB2, Informix, Microsoft SQL Server, Oracle or Sybase servers.

JNDI

Java Naming and Directory Interface 1.2 is a major upgrade that includes support for event notification and LDAP, Version 3 extensions and controls. WebSphere Application Server supports JNDI 1.2 today for EJB lookup, as well as JNDI over LDAP.

RMI-IIOP

RMI-IIOP speeds distributed application development by permitting developers to work completely in Java technology. When using RMI-IIOP to produce Java technology-based distributed applications, there is no separate IDL or mapping to learn. Like RMI, this provides flexibility by enabling developers to pass any Java object that can be serialized (referred to as objects by value) between application components. Like CORBA, RMI-IIOP is based on open standards defined with the participation of hundreds of vendors and users in the Object Management Group (OMG). Like CORBA, RMI-IIOP uses IIOP as its communication protocol. IIOP eases legacy application and platform integration by enabling application components written in C++, SmallTalk and other CORBA-supported languages to communicate with components running on the Java platform.

With RMI-IIOP, developers can write remote interfaces in Java and implement them just using Java technology and the Java RMI APIs. You can implement these interfaces in any other language supported by an OMG mapping and a vendor-supplied ORB for that language. Similarly, clients can be written in other languages using IDL derived from the remote Java technology-based interfaces. Using RMI-IIOP, objects can be passed both by reference and by value over IIOP. It combines the usability of Java RMI with the interoperability of the IIOP. It provides you with a powerful environment in which to perform distributed programming in Java. WebSphere Application Server fully supports RMI-IIOP for communication.

Security

WebSphere Application Server supports secure integration with the Web server (HTTP/SSL), and also supports SSL directly to the application server for Java clients (RMI-IIOP/SSL). WebSphere also supports third-party authentication (including a lightweight version) for secure delegation. Support for Public Key Infrastructure (PKi) is provided in the form of authentication through digital certificates. Enhanced in Version 5.0 is support for Kerberos, offering strong, standards-based authentication security for client/server applications that use secret key cryptography.

WebSphere Application Server supports distributed registries for authentication, method level authorization, single sign-on, existing pluggable registries, LDAP schema, secure delegation between servers and distributed authorization (ACLs) for EJB, servlets, JSP and HTML content. Java Authentication and Authorization Service (JAAS) extends the security architecture of the Java 2 platform with additional support to authenticate and enforce access controls upon users. JAAS is the latest development in the Java Security Architecture, building upon the security package in the core SDK, the Java Secure Sockets Extension (JSSE), the Java Cryptography Extension (JCE) and the Java Security Tools. JAAS enables developers to authenticate users and enforce access controls upon those users in their applications. It simplifies application development by serving as a building block for developers and minimizes the risk of creating subtle but dangerous security vulnerabilities in application code. JCE provides a framework and implementation for encryptions, key generation and key agreement and Message Authentication Code (MAC) algorithms. Web services gateways are also implemented in Version 5.0, offering more security and protection by filtering Web services access to registries and other applications. Finally, Common Secure Interoperability, Version 2 (CSIv2) is supported in Version 5, offering higher security levels and access protection for EJB components and CORBA applications.

XML/XSL

WebSphere Application Server supports the latest specifications — Java API for XML Programming (JAXP), Document Object Model (DOM) and Simple API for XML (SAX). IBM is an industry leader in XML, and WebSphere Application Server includes complete XML support for parsing and understanding information sources that utilize XML tags. WebSphere Application Server, Version 5 includes complete XML development and parsing support for deployment. For more information about IBM XML support, including downloads of XML parsers and tools, see the IBM developerWorks XML zone.

Summary

The latest IBM WebSphere software platform-based technologies—WebSphere Application Server, Version 5.0 and WebSphere Application Server, Version 5 for z/OS and OS/390—focus on enhanced support for leading open standards, encompassing those required for Java 2 Enterprise Edition connectivity, as well as those needed to extend applications for business-to-business Web services. The specifics described in this paper relay how IBM WebSphere software platform implements these standards according to the spirit of their intent, creating a development and deployment environment that rapidly grows and adapts to meet your toughest real-world application demands. As standards continually evolve, you can visit the following related Web sites to get the most up-to-date WebSphere software platform and platform-based technology information.

For more information

To learn more about IBM WebSphere software platform and platform-based technology, visit:

- *IBM WebSphere home page at* software.**ibm.com**/websphere
    - *View product information.*
    - *Click WebSphere Developers Domain for more technical topics.*
- *IBM developerWorks at* **ibm.com**/developer
- *IBM WebSphere Studio development environment at* **ibm.com**/websphere/studio
- *IBM Software Strategy for e-business at* **ibm.com**/software/ebusiness
    - *Also, get access through IBM WebSphere Developer Domain.*
- *IBM Redbooks™ at* redbooks.**ibm.com**
    - *Also get access through IBM WebSphere Developer Domain.*
- *Java 2 Platform, Enterprise Edition—Overview of Technologies (Sun Microsystems site) at java.sun.com/j2ee*
- *Java 2 Platform, Enterprise Edition—Downloads and Specifications (Sun Microsystems site) at java.sun.com/j2ee/download.html*
- *Web services programs at IBM—***ibm.com**/webservices
- *Web services on WebSphere (WoW)—***ibm.com**/websphere/wow
- *WebSphere Technology for Developers download (through WebSphere Developers Domain)* **ibm.com**/websphere/developer

IBM®

e business software