



WebSphere. software

Robust Infrastructure for SOA

**A Technical Introduction to
IBM WebSphere Application Server V6.1**



Contents

Contents	1
Executive Overview	2
The Evolving Role of Application Servers	3
WebSphere Application Server Family Overview	5
Simple, Rapid Development and Deployment	9
A Highly Integrated Development and Deployment Environment.....	9
Build Quickly; Expand Easily	11
Maximize the Value of Development Assets and Investments.....	13
Develop Dynamic Applications	14
Efficient Deployment.....	14
Secure, Scalable, Highly Available SOA Runtime	17
Instill Confidence with Security-Rich Features	17
Always On, Always Available	19
Continuous Availability and High Performance Computing.....	20
Enhanced User Experiences	22
Harness the Power of z/OS	23
Extensive Communication Services	25
Web Services Standards	25
Session Initiation Protocol (SIP).....	26
Integration of Asynchronous Messaging Services	26
Service Integration Bus Architecture	27
Integrate and Reuse Existing IT Assets	28
Agile Administration and Management	31
Efficient Deployment.....	31
Effective Application Management	32
Extended Manageability with WebSphere Extended Deployment	34
Performance Monitoring	35
A Leading Technology Partner	37
For More Information	38

Executive Overview

IBM WebSphere® Application Server, Version 6.1 is the foundation of the IBM WebSphere software platform and is a key building block for a Service Oriented Architecture (SOA). As the leading Java™ 2 Platform, Enterprise Edition (J2EE) 1.4 and Web services application platform, WebSphere Application Server delivers a high performance transaction engine and extremely scalable, highly available, and secure platform for SOA.

With the same programming model and core application server functionality across configurations¹, WebSphere Application Server, Version 6.1 accelerates business flexibility. By enabling enterprises to standardize, automate, and integrate services, clients can maximize return on investments while responding, on demand, to changing business opportunities.

With WebSphere Application Server, you can grow seamlessly from basic single server implementations to more sophisticated distributed topologies that leverage multi-server clustering for on demand scalability and high availability, all on a common server foundation. WebSphere Extended Deployment extends this foundation, enabling a business-grid through a dynamic, goals-directed, high performance application environment for running mixed application types and workload patterns in WebSphere. IBM WebSphere Application Server is a proven, agile platform designed to support today's urgent business imperatives. WebSphere Application Server can help you reduce overall costs, improve customer loyalty and respond quickly to new business opportunities. WebSphere Application Server delivers the following advantages:

- **Simple, rapid development and deployment** - WebSphere Application Server, Version 6.1 enables rapid time to market, and increased productivity for the individual developer with support for rapid development frameworks and extensive automation. Streamlined packaging, simplified installation, and the rapid deployment framework speed time to deployment.
- **Secure, scalable, highly available SOA runtime** – WebSphere Application Server, Version 6.1 can provide a secure, scalable, and highly available environment in which to run services. With support for advanced workload management, dynamic caching, dynamic clusters and high availability management, WebSphere Application Server Network Deployment and WebSphere Application Server for z/OS provide near continuous uptime for mission critical applications.
- **Extensive Communication Services** – WebSphere Application Server, Version 6.1 enables reuse of existing assets with standards-based communication, messaging and comprehensive support for Web services. The result? Increase in Return on Investment (ROI) and lower Total Cost of Ownership (TCO).
- **Agile administration and management** – Simplified and unified administration across all configurations of WebSphere Application Server, Version 6.1, minimizes training costs and makes it much easier to migrate across these WebSphere Application Server configurations.

¹ This statement is limited to the WebSphere Application Server *Version 6.1* configurations, and *does not include* WebSphere Application Server Community Edition. IBM is targeting to make WebSphere Application Server – Express V6.1 available before the end of 2006. All statements regarding IBM's plans, directions, and intent are subject to change or withdrawal without notice.

As an affordable entry point to SOA, WebSphere Application Server promotes re-use and facilitates connectivity. It provides a flexible, secure SOA infrastructure that can deliver real business value. With WebSphere Application Server you can:

- Build and deploy application services quickly and easily, for improved time-to-value
- Run services in the most secure, scalable, highly available environment, for improved service availability and peace-of-mind
- Re-use software assets and extend their reach, for maximum business flexibility
- Manage applications effortlessly, for minimized IT operational costs
- Grow as needs evolve, leveraging the assets and skills developed, for increased ROI

WebSphere Application Server provides features to support dynamic applications, improved ease of management and the latest J2EE/Java 2 Platform, Standard Edition (J2SE) technology and Web services standards. WebSphere Application Server, Version 6.1 continues to build on its solid reputation adding new capabilities that meet the needs of today's business leaders. With IBM and the award-winning family of WebSphere products, you can fully address the challenges of a rapidly changing business world.

The Evolving Role of Application Servers

Application servers have evolved rapidly from their early days as products that provided relatively simple component runtimes (and often as simply an extension to web servers providing dynamic content to web applications). Today they provide a comprehensive infrastructure for running applications that connect people and processes, providing the underlying services for portal, business integration, and commerce solutions. They provide a secure, scalable and reliable common environment and programming model for running the applications that run your business.

Service Oriented Architecture (SOA) helps customers increase the flexibility of their business processes, strengthen their underlying IT infrastructure and retain and reuse their existing assets. IBM WebSphere Application Server, Version 6.1 has become the foundation building block for SOA applications that are easy to use, interoperable, secure and transactional. The application server thus has become the core enabler of a standards-based approach to SOA for building business flexibility. As the application server has matured into core technology for an SOA platform, it is increasingly called upon to fulfill a variety of roles within organizations, at a variety of scales, and with customers at different stages of SOA adoption. To truly enable business flexibility, it is important that the application server addresses all these roles with a common programming model that allows you to respond on demand by scaling up and scaling out (across heterogeneous environments) quickly and easily. Support for clustering and high availability management ensures this can be done with near continuous uptime for mission critical applications.

While many customers require robust application infrastructure features, like those found in WebSphere Application Server, V6.1, others are deploying applications on lighter-weight open source application servers. With the addition of WebSphere Application Server Community Edition (a J2EE application server built on open source technology from the Apache Software Foundation) to the WebSphere Application Server family, customers can deploy to the application servers that best meet their objectives.

WebSphere Application Server Community Edition V1.0.1 introduces a new business model to IBM's market-leading WebSphere Application Server family. Providing code that is free to download and use along with a range of for-fee technical support options brings increased choice and flexibility. Customers can begin development, testing and deployment at no cost. As their needs grow, customers can evolve to the more advanced WebSphere Application Server products with minimal disruption since WebSphere Application Server Community Edition V1.0.1 is based on the J2EE programming model, and provides an Eclipse plug-in with J2EE profiling to ensure that the applications created are portable. Not only do customers have the flexibility to blend open source business models with the reliability, scalability and resilience of the traditional WebSphere products, they can select the qualities of service that best meet their development and deployment strategy, knowing that their technology investments will be preserved as their needs evolve and grow.

In summary, an application server platform must provide consistently, an infrastructure that maximizes flexibility and reduces the complexity of building, deploying, and managing services. It must also provide the ability to achieve near continuous availability and near linear scalability on demand. WebSphere Application Server, Version 6.1 addresses these needs by delivering:

- Simple, Rapid Development and Deployment
- Secure, Scalable, Highly Available SOA Runtime
- Extensive Communication Services
- Agile Administration and Management

This paper outlines these requirements in more detail and describes how WebSphere Application Server, Version 6.1 addresses them to deliver a highly scalable, highly available platform for SOA that delivers benefits in terms of reduced costs, reduced time to value, and increased business flexibility.

WebSphere Application Server Family Overview

WebSphere Application Server, Version 6.1 is the foundation of the IBM WebSphere software platform and is a key building block for a Service Oriented Architecture (SOA). As the leading J2EE 1.4 and Web services application platform, WebSphere Application Server, Version 6.1 delivers a high performance transaction engine and extremely scalable foundation for traditional or SOA development.

WebSphere Application Server, Version 6.1 is available in multiple configurations (Figure 1) across a wide variety of hardware and operating systems. Importantly, WebSphere Application Server, Version 6.1 delivers the same full J2EE 1.4 and J2SE 5.0 programming model and core application server functionality across configurations¹. With WebSphere Application Server, Version 6.1 you can grow from basic single server implementations to more sophisticated distributed topologies that leverage multi-server clustering for on demand scalability and high availability, all on a common server foundation. This provides customers with the flexibility to:

- Choose the level of capability best suited to current needs
- Expand, scale and grow as business needs evolve
- Respond on demand to the changing marketplace without the cost of re-architecting, reprogramming, and migrating to different technology bases

With unbeatable platform support, WebSphere Application Server, Version 6.1 delivers the greatest flexibility in deployment choices, including support for: Windows®, Linux® on many platforms (from IA-32 to System z®), IBM AIX®, Sun Solaris®, HP/UX®, IBM OS/400® and i5/OS™, z/OS®, and selected 64-bit platforms (now includes AIX and zLinux). Customers routinely run WebSphere software across a variety of hardware and operating system platforms. Companies can pilot their applications on Microsoft Windows and later deploy and scale to their platform of choice as business needs dictate.

WebSphere Application Server, Version 6.1 provides a strong migration path for existing customers with the ability to run older J2EE 1.3 and 1.2 applications as well as the new J2EE 1.4 and J2SE 5.0 Java and SOA applications. In addition, support in WebSphere Application Server, Version 6.1 (Network Deployment and z/OS configurations) for mixed V5 and V6 server environments, provides for co-existence and a more flexible and staged migration to Version 6.1.

WebSphere Application Server, Version 6.1 offers multiple entry points with the right value to most closely match your needs today—while maintaining the flexibility to grow and migrate by scaling up and scaling out. WebSphere Application Server supported growth scenarios include:

- Extending or migrating to new hardware and operating system platforms
- Upgrading to more sophisticated configurations of WebSphere software
- Consolidating operations from other application servers or consolidation of physical servers

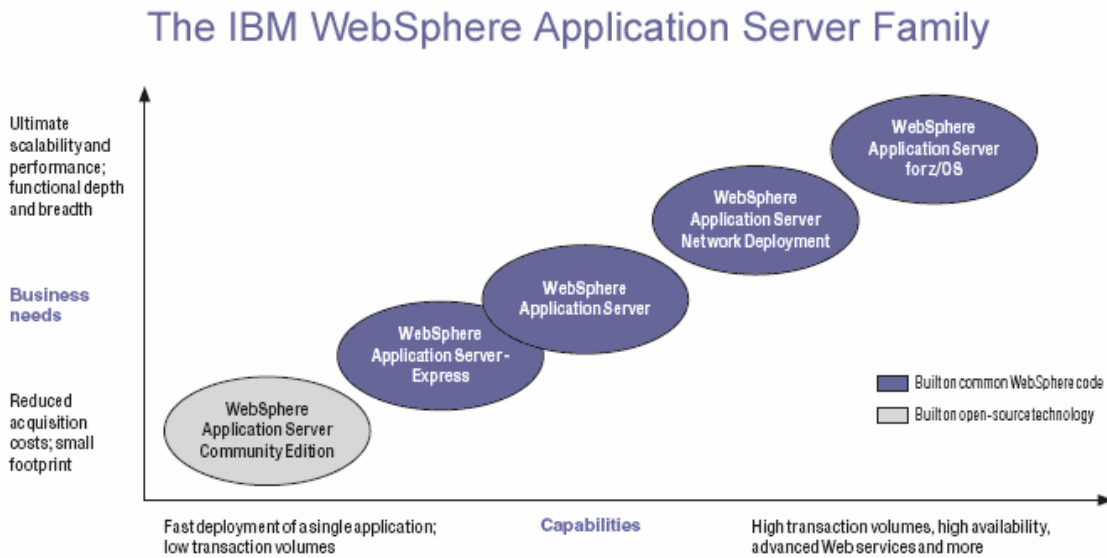


Figure 1 - WebSphere Application Server Configurations

The different WebSphere Application Server configurations provide customers with clear entry points and migration paths from one configuration to another as business needs grow. All configurations deliver a solid foundation building block for SOA that integrates enterprise data, transactions, and Web services and include the IBM Tivoli Access Manager and Tivoli Directory Server for WebSphere Application Server (an LDAP server) for centralized identity management and security in enterprise distributed environments¹.

WebSphere Application Server Community Edition V1.0.1² is a lightweight J2EE application server, built on open source, Apache Geronimo technology. Designed to help you accelerate your development and deployment efforts, it harnesses the latest innovation from the open source community to provide a readily accessible and flexible foundation for building Java applications. WebSphere Application Server Community Edition is backed by world-class support from IBM (purchased separately), assuring you of reliable assistance and timely problem resolution from IBM's team of expert support professionals and developers.

WebSphere Application Server - Express is the entry point for customers that require a rapid and affordable solution for easily developing, deploying, and managing applications on a single server with up to two processors. The Express configuration delivers an out-of-the-box solution for managing simple yet dynamic Web sites with an easy-to-use application server and development environment. In common with the other WebSphere Application Server configurations, WebSphere Application Server – Express supports the entire J2EE 1.4 programming model, including Enterprise Java Beans (EJB) and the Java Connector Architecture (JCA). The Express configuration includes a visual application development tool, as well as a set of integrated applications, wizards and samples to help you get up-and-running, and productive, quickly.

² The focus of this whitepaper is WebSphere Application Server V6.1, and not WebSphere Application Server Community Edition For more information on WebSphere Application Server Community Edition, please go to: <http://www-306.ibm.com/software/webservers/appserv/community/>

WebSphere Application Server is optimized to ease administration in a scalable, single-server deployment environment and is perfect for stand-alone, departmental applications and Web services. This core configuration offers unlimited processor support, simple HTTP failover / load distribution, and additional platform support, such as Linux on System z.

WebSphere Application Server Network Deployment is the primary entry point for enterprises that need higher scalability and quality of service built in. It extends the WebSphere Application Server base package to include clustering capabilities, edge-of-network components, centralized management and monitoring, extended web services support, and high availability for distributed configurations. This is the configuration for distributed, clustered, highly available, high volume environments. In addition to the base capabilities of the WebSphere Application Server, WebSphere Application Server Network Deployment delivers enterprise scalability and availability through intelligent workload management (WLM), extended Web services support through the Web services Gateway, edge-of-network caching services, autonomic high availability management (HAManager), backup cluster support, central management across multiple clustered application servers, and support for Network Attached Storage shared file system technologies. WebSphere Application Server Network Deployment, Version 6.1 provides significant high availability features out of the box, such as WLM-mediated failover at the HTTP, web container, and EJB container levels, to help ensure 24x7 availability.

WebSphere Application Server for z/OS delivers a J2EE application server optimized to utilize the unique qualities of service provided by IBM System z9™ hardware and the z/OS operating system. It is the entry point for customers that need to leverage the unique resilience and qualities of service of the System z9 platform, or leverage the TCO benefits of server consolidation on System z9. WebSphere Application Server for z/OS, Version 6.1 provides the ability to achieve higher performance and dramatically lower TCO by exploiting the System z Application Assist Processor (zAAP), a dedicated processor for Java workloads running under the IBM JVM and free from processor cycle charges. This allows you to offload processing cycles from the general purpose processor, improving overall performance while reducing TCO. With the same full J2EE 1.4 and J2SE 5.0 programming models as WebSphere Application Server Network Deployment, WebSphere Application Server for z/OS brings qualities like close proximity to enterprise data, prioritized workload management, complex transactional integrity, intense scalability and rich mainframe security, to Java and Web services applications.

WebSphere Extended Deployment is an add on to existing WebSphere Application Server Network Deployment and WebSphere Application Server for z/OS environments that extends them to provide on demand dynamic responsiveness, high performance computing enhancements, simplified administration and management of complex distributed environments and a flexible approach to support both transactional and long running workloads. As a simple delta to these existing WebSphere Application Server installations, WebSphere Extended Deployment does not require migration or restructuring of the existing environment and is integrated into the WebSphere administration environment. WebSphere Extended Deployment extends capabilities to dynamically accommodate variable business demands, support mixed workloads (transactional and long running) and mixed server environments, and reduce the complexity and cost of managing complex distributed environments. WebSphere Extended Deployment delivers extended capabilities in four key areas:

Robust Infrastructure for SOA

- *Flexibility* - Provides support for mixed application types and workload patterns in WebSphere, enabling you to better leverage your existing application infrastructure resources.
- *Reliability, scalability, high performance* - Enhances quality of service of business critical applications to support near linear scalability for high-end transaction processing, helping you improve customer service levels while also leveraging existing Java skills and resources.
- *Dynamic scaling* - Allows your application environment to scale on demand with virtualization of WebSphere resources and use of a goals-directed infrastructure, helping you increase the speed at which your company can adapt to business change.
- *Manageability* - Offers simpler and improved management of complex system operations with real-time operational health monitoring tools, helping you reduce the cost of managing IT resources.

Together, these extended features of WebSphere Extended Deployment allow your application environment to scale on demand with improved resource utilization, lower TCO, autonomic management and continuous high availability.

The WebSphere Application Server family provides a consistent platform for SOA, from Express to Network Deployment and z/OS. These configurations provide consistent support for the full J2EE 1.4 and J2SE 5.0 programming models and latest Web services standards, core WebSphere programming model extensions, and feature broad platform support. This consistency and coordination among WebSphere Application Server configurations enables application portability across platforms. As well, it enables the flexibility to migrate and scale easily to other configurations when more advanced capabilities are needed, such as higher availability and advanced workload management, or the unique qualities of services provided by the z/OS platform, without the cost of re-architecting, reprogramming, and migrating to different technology bases. And with tight integration across the IBM software portfolio, you are positioned for rapid business growth.

Simple, Rapid Development and Deployment

WebSphere Application Server and IBM Rational tools for WebSphere provide integrated visual development tools and dynamic frameworks to help you leverage existing skills and assets, and build agile applications. With an open-service-oriented architecture, you can easily integrate new and existing assets, increase your business flexibility and facilitate the efficient use of your IT resources. With WebSphere Application Server, Version 6.1 and the Rational Software Development Platform, you have a comprehensive build-to-integrate foundation for SOA applications.

In addition to the Application Server Toolkit (AST) found within WebSphere Application Server, tools from IBM Rational provide an easy to use environment for the many different developer roles throughout the application lifecycle. With WebSphere Application Server, Version 6.1 you have an integrated application development and deployment platform that can positively impact your return on investment by:

- Enabling rapid time to market for new applications, from concept through debug and test into production—regardless of the deployment server platform or operating system
- Facilitating the reuse of existing assets and skills when creating new applications
- Building in flexibility that makes it easy to expand and adapt applications as business needs change
- Driving productivity for the individual developer with support for rapid development frameworks and extensive automation — and within the development team by supporting different developer roles throughout the organization

A Highly Integrated Development and Deployment Environment

WebSphere Application Server, Version 6.1 includes the Application Server Toolkit (AST) which provides substantial, integral development and deployment capabilities for SOA. Additionally, WebSphere Application Server is tightly integrated with the IBM Rational Software Development Platform, which provides a powerful and flexible foundation for business-driven development for SOA. The IBM Rational Software Development Platform is a framework of integrated tools and best practices that support a proven end-to-end process for all developer roles in the application development life cycle.

The AST is based on the familiar Eclipse workbench technology and provides features for team programming, local and remote debugging, J2EE application deployment and more. The AST allows you to create, assemble, and deploy tasks for web applications such as importing WAR files, creating web projects, copying code artifacts from one Web project to another and editing HTML, JavaScript and XML. The AST also supports assembly tasks for enterprise applications such as importing EJB JAR files, creating new EJB projects, copying code artifacts, and generating EJB deployment code. Comprehensive support for unit testing applications is included, providing all the functionality necessary to deploy and debug applications, including a Unit Test Client Web application for easy testing of EJBs and Web services. The AST includes the rapid deployment feature, providing an easy mechanism to deploy applications to a running WebSphere Application Server Version 6.1 instance.

With WebSphere Application Server, Version 6.1, the AST has been updated to provide support for new capabilities including: tools to create JSR 168 compliant portlets; new administration tools for the creation and maintenance of wsadmin Jython files; and tools to edit WebSphere specific bindings and extensions.

The AST also provides the tools necessary to develop and export JSR 116 Session Initiation Protocol (SIP) servlets (see the Extensive Communication Services chapter for more on SIP), including wizards to create SIP services, a rich Deployment Descriptor Editor, and import/export wizards to package and deploy SIP Servlets. New Automation tools provide a full-scale Integrated Development Environment for developing and debugging WebSphere management Jython scripts offering features like color-coded source display, command completion, configuration navigation, and syntax checking. A set of common administration tasks are also provided in the tools allowing you to capture actions completed in the admin console, and generate the equivalent Jython commands into the script being edited.

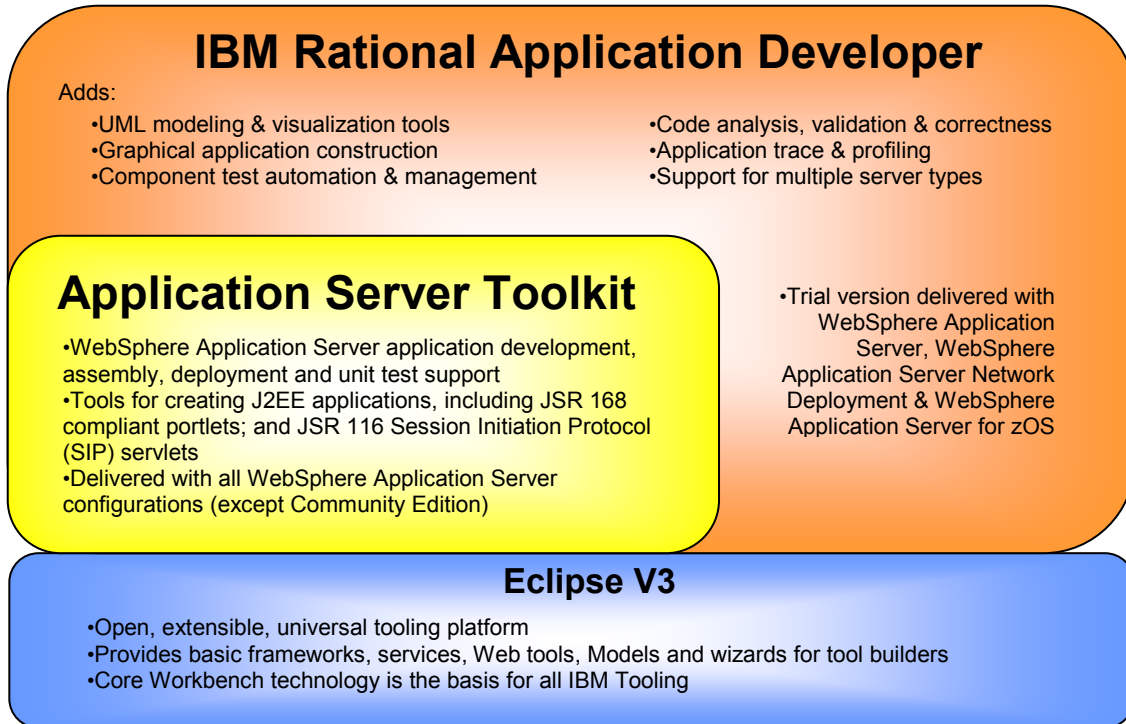


Figure 2 - Integrated development and deployment tools for WebSphere Application Server V6.1

The Application Server Toolkit and Rational Application Developer share in common an Eclipse based Integrated Development Environment (IDE) and are the principal tools for developing WebSphere SOA applications. As such, the AST contains a sub-set of Rational Application Developer/Rational Software Architect functionality (Figure 2). Existing AST projects can be imported to Rational Application Developer and Rational Software Architect to take advantage of the added capabilities. While the Rational tools are available separately, this tools platform for WebSphere Application Server provides extensive automation and visual tools for building new services and composing new applications out of a set of enterprise services—whether they are Web services, EJB applications or legacy resources. Both tools provide an integrated WebSphere runtime test environment, speeding iterative development and testing and enabling developers to build in quality at all stages of development. The result for your company is higher quality, lower cost, more flexibility and faster time to value. Built on the Eclipse V3 open systems development platform, Rational tools and the AST provide open tools integration and the ability to adapt and extend the

development environment with best-of-breed plug-in tools from IBM, IBM Business Partners and the Eclipse community at large.

Build Quickly; Expand Easily

The time required to roll out new applications is a key concern throughout all industries, and improved developer productivity is clearly a way to address this concern. One way to vastly improve productivity is to reduce the need for manual programming. You can do this through powerful frameworks that absorb much of the work involved in development, pre-built widgets that can be exploited to reduce custom development, or through tools that generate code used by the runtime. With SOA, developers are tasked with creating Web services or developing systems that consume Web services. Doing this manually involves a substantial amount of tedious and potentially error-prone work. With IBM's developer tools for WebSphere, developers can focus on writing the business logic, while the tools (e.g. Rational Application Developer, Figure 3) automate everything from the Web Services Description Language (WSDL) file and code generation to test-client generation and Web Services-Interoperability (WS-I) conformance verification.

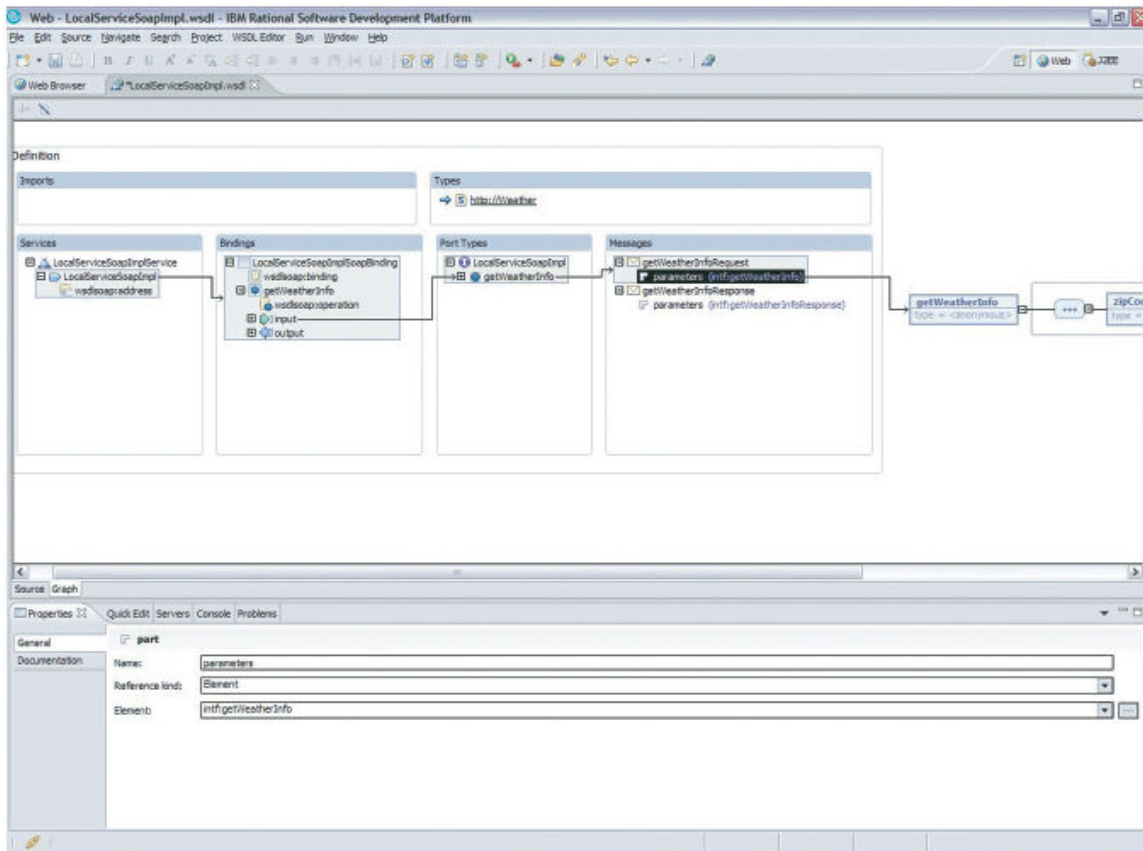


Figure 3 - IBM Rational Application Developer – automates creation of Web services from existing assets (e.g. JavaBeans, EJBs, etc.)

WebSphere software delivers a combination of frameworks and tools that work together to provide best practice implementations, a realization of the industry's best architectures. With the Rational development tools for WebSphere, developing and maintaining applications is simplified through more visual

programming, and use of standards-based frameworks. For example, user interface developers can use JavaServer Faces (JSF) to visually wire together JSF UI components. Included with WebSphere Application Server Version 6.1 is a JSF Widget Library (JWL) that helps save development time and expense by providing a set of pre-built functions for use within Web user interfaces. Developers can quickly add features like graphs, menus, data-grids, and calendar date-pickers quickly, making web user interfaces richer with minimum effort. Similarly, a FacesClient Framework provides client-side processing coupled with a client-side data model to sustain longer end user interactions without roundtrips back to the server. This reduces server and network loads and user wait times resulting in lower consumption of server-side resources.

Introduced with WebSphere Application Server Version 6.1 is support for Java 2 Standard Edition (J2SE) 5.0, delivering substantial performance improvements and providing developers with an unparalleled development environment that can increase application programmer productivity. Developers can use many new API libraries, including generics, auto boxing of primitives, annotations, and enumerated types. Annotations provide a new metadata feature that allows the association of additional data alongside interfaces, methods, fields, and Java classes, which can be read by tools and if configuration permits, can be stored in the class file and discovered at runtime. In this way, tools used to develop J2SE applications can leverage common infrastructures, potentially reducing the resources needed to program and deploy applications, resulting in more efficient development and modification of applications.

Also introduced in WebSphere Application Server, Version 6.1 is the ability to write portlets, in addition to servlets. Administrators can configure, manage, and run portlet applications without the requirement for a full Portal Server. Users can access the portlets with URLs, as they do servlets. This movement of the JSR 168 (specification that enables interoperability among portlets and portals) portlet container from IBM WebSphere Portal down into WebSphere Application Server, Version 6.1 provides a common programming model for presentation logic across all of IBM's application server-based offerings. This also means that you no longer need to choose between servlet and portlet programming models when creating a presentation layer, future proofing your applications and thus eliminating the need to potentially transform or rewrite presentation logic when moving from one WebSphere Application Server offering to another.

Developers can visually connect Java applications to enterprise information systems using JCA integration components or as Web services. Service Data Objects (in the standardization path of the Java Community Process) provide a unified framework for connecting to disparate data sources rapidly and easily. Overall, Rational tools for WebSphere help minimize the amount of time and effort needed to create and deploy new application functionality. They provide features that simplify SOA development and maximize developer productivity, including:

- Extensive automation and generation, for the core elements of J2EE and Web services, with automatic WS-I conformance verification.
- Automatic creation of Web services from existing assets (e.g. Java classes, EJBs, or EIS – JCA assets) enabling rapid reuse and integration within new SOA applications.
- Support for rapid development frameworks, such as JavaServer Face, and Service Data Objects which provide a unified programming model for data access across disparate data sources and protocols. This simplifies development and improves flexibility.

- Support for Unified Modeling Language (UML) model driven development “top-down”, with generation of Web services elements from requirements specifications and UML models. (e.g. generation of WSDL files and XML schema to describe the Web services, and JavaBeans or EJB skeletons to implement the services).
- Integrated WebSphere Application Server runtime test environment, with automatic generation of Web services test clients, stubs, and harnesses. This enables you to test the new services and applications immediately and implement subsequent changes without disruption to the running production applications. The integrated tools and server enable a tight develop, test and debug cycle for both functional and performance testing by each developer.
- The rapid deployment feature, which simplifies both development and deployment experiences via support for annotation based programming and automating deployment including change detection management.

As a core (and included) part of WebSphere Application Server, the Application Server Toolkit provides substantial integral development and deployment capabilities. The AST provides J2EE assembly and deployment perspective tools, including the rapid deployment feature, which dramatically simplifies application deployment and ongoing deployment management. The Rational tools provide extended support for rapid development of new applications. These many rapid development and deployment features deliver real business benefits in terms of reduced development cycle times, maximization of existing skills and resources. They free-up developers to focus on implementing the business logic of the services they are developing, and build in quality early in (and throughout) the development cycle.

Maximize the Value of Development Assets and Investments

Building new applications that integrate multiple back-end systems and maximize existing assets typically requires data transformation and transactional integrity. WebSphere Application Server, Version 6.1 delivers productivity through an open approach to transforming any application asset into a modular, network-accessible service, which others can easily identify and reuse. With support for Java API for XML-based RPC (JAX-RPC) 1.1, developers can map and create interoperable and portable Web services from existing Java assets. For packaged applications, enterprise information systems, and other legacy assets, full support for the Java Connector Architecture (JCA) 1.5 allows easy integration as either a JCA resource managed by WebSphere Application Server, or as a reusable and interoperable Web service.

Most companies are not starting from a “blank page” as they move forward to standardize their business processes and increase business flexibility with SOA technology based applications. In most cases, they have significant investments in existing systems that they need to leverage in combination with newer business logic written in Java as they migrate to a SOA. Comprehensive analysis of existing legacy assets is critical to successfully leveraging these assets for reuse. IBM WebSphere Studio Asset Analyzer is a powerful tool that can be used to analyze existing application assets from Web pages, to Java components, to host assets, such as COBOL, PL/1 and JCL. Using the knowledge store built from the analysis of assets, WebSphere Studio Asset Analyzer helps identify reusable components, helps you understand the impact of changing them and helps prepare them for broader use as services accessible by new and existing business

applications. As a result, you can maximize the value of existing assets that have been running your business for years (if not decades) and continue to leverage the skills and experience of all your developers.

Develop Dynamic Applications

WebSphere Application Server simplifies the development of dynamic applications through an industry-leading J2EE technology and Web services-based application server platform that provides advanced programming model extensions. It allows a business to create new opportunities by exposing application services for integration by other businesses, organizations or platforms. WebSphere extensions, such as the Internationalization Framework and application profiling capabilities provide strong separation between underlying development infrastructure and the runtime deployment experience of the applications. This functional abstraction enables dynamic applications that can respond and adapt automatically to global constituencies to be easily modified as business conditions and policies change, without modifying the underlying code of the application. Similarly, Service Data Objects provide a unified programming model for data access, that decouples the business logic code of a WebSphere application from the specifics and APIs of the target back end resource. This not only improves development productivity but provides a more flexible architecture for accommodating change. WebSphere Application Server Version 6.1 provides an EJB Service Data Objects (SDO) Mediator to simplify the programming model further. Where current techniques for implementing disconnected data objects involve a combination of copy helper objects, session beans and EJB access beans, the use of the EJB mediator reduces the amount of programming. Dynamic data objects provide flexibility and eliminate the need to define copy helper type objects. Increased performance can be achieved with optimized queries and having the EJB mediator read and write directly to the data store, bypassing the need to activate EJB instances.

Other features of WebSphere Application Server, Version 6.1 contribute to dynamic application flexibility. The Dynamic Query Service (a WebSphere programming extension) lets you dynamically build and submit queries and perform calculations on application data at run-time. This ability to pass in and process EJB query language queries at run-time, eliminates the need to hard-code required queries into deployment descriptors during application development. This not only speeds development (by allowing interactive testing) but results in a more flexible, dynamic, and maintainable application. These dynamic application adaptability features deliver clear business benefits and also help eliminate months of programming costs and significant IT maintenance costs.

Efficient Deployment

Deploying WebSphere Application Server and applications continues to become easier than ever. With Version 6.1, IBM continues to deliver enhancements to improve the deployment from the initial out-of-the-box experience, through ongoing deployment and administration of highly distributed and clustered applications.

WebSphere Application Server provides streamlined packaging and simplified installation with a single CD install image for the application server per platform. With WebSphere Application Server, Version 6.1, non-root users can perform full installation and removal, incremental feature installations and edition updates. IBM Installation Factory for WebSphere Application Server, Version 6.1 provides enhanced facilities to make server installation and deployment easier, more reliable and repeatable. Support for installation profiles

allows you to install one copy of the core files and then roll out multiple application server runtimes based on that common profile. New to Installation Factory for WebSphere Application Server, Version 6.1 is the ability to incorporate and bundle fix packs, applications (one or more EAR / enhanced EAR files) and their configurations, and any relevant scripts (which can be executed every time a new profile is created and not just during installation) into the custom install image. You can now also generate cross-platform install packages that can be deployed to the UNIX/Linux and Windows platforms, saving you time that can be focused on real business issues.

WebSphere Application Server also provides simplified and rapid application deployment facilities significantly reducing the complexity of building J2EE applications. Through an easy to use framework, WebSphere Application Server's rapid deployment feature automates the most common aspects of creating, assembling, configuring / packaging, and deploying J2EE applications and their myriad of artifacts (e.g. Java source files, Java class files, images, XML, HTML, etc.). The rapid deployment tools free you from having to understand J2EE application structure and dramatically simplify the process of installing, modifying, and uninstalling applications on a WebSphere Application Server.

The rapid deployment feature is a collection of Eclipse plug-ins to the AST and Rational Application Developer environments. The rapid deployment feature extends these tools to support annotation-based programming and deployment automation. Annotation based programming allows developers to maintain a single Java source file with embedded metadata "annotations", and automatically generate the multiple source files and artifacts on deployment using those metadata (Figure 4). Deployment automation provides automated "file copy like" installation as well as automated change detection management and regeneration of required artifacts as needed (Figure 5).

Single Java source file with annotation-based programming

```
package com.ibm.wrd;
/**
 * @ejb.session name="Hello" type="Stateless"
 * view-type=both jndi-name="HelloBean"
 */
public class Hello
{
/**
 * @ejb.interface-method view-type=both
 */
public String hello(String name)
{
return "Hello: " + name;
}
}
```

Hello.java

Multiple Java source files and application artifacts

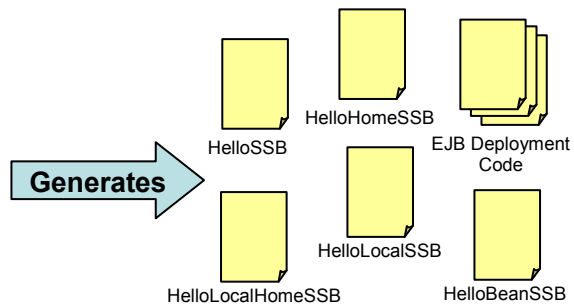


Figure 4 - Annotation based source coding simplifies development and deployment

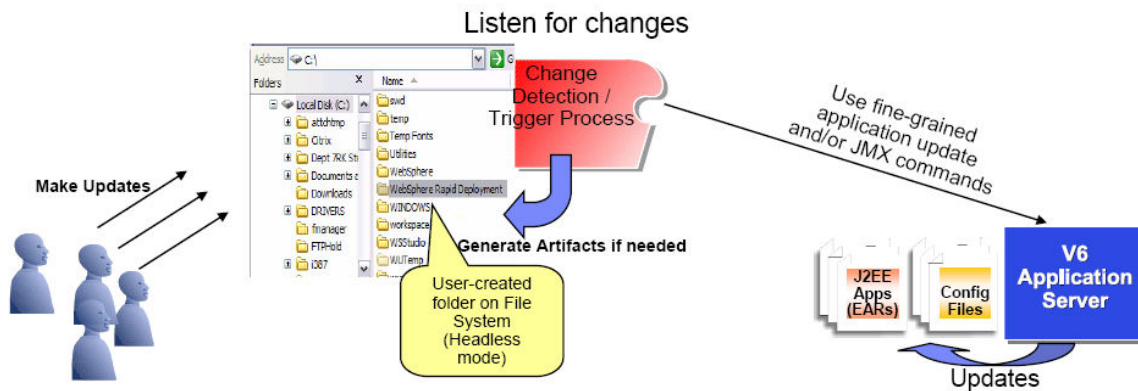


Figure 5 - Automatic installation and change detection simplifies deployment and management

These advanced development and deployment features enable customers to improve time to value and make the most of existing technology skills, by:

- Simplifying the development experience for WebSphere applications by reducing the number of development artifacts, concepts, and technologies; and by providing support in established familiar tools for the development and deployment model.
- Simplifying the deployment experience for WebSphere applications by automating the process of installing an application on WebSphere, reducing the amount of information that must be collected by the installer, and automating the process of activating incremental changes to an application on a running server.

Secure, Scalable, Highly Available SOA Runtime

WebSphere Application Server, Version 6.1 provides a secure, dynamic platform that allows you to do more work with fewer resources. In today's volatile business environment, the ability to continually optimize your network for enhanced performance, scalability and availability has a direct impact on the top and bottom lines of your business. High-volume, dynamic Web sites place significant demands on an SOA infrastructure. Attracting and keeping customers who have a choice to go elsewhere requires reliable and secure access to business-critical applications. And refreshing frequently requested data must occur dynamically and transparently so the Web sites are available without interruption. All these functions must happen while continuing to drive efficient operations and keep costs down.

WebSphere Application Server leverages proven experience to deliver a scalable, highly available and security-rich platform. Through performance and availability features, together with edge-of-network technology and advanced security capabilities, WebSphere Application Server, Version 6.1 can dynamically and more securely react to network pressures. WebSphere software provides an intelligent, optimized application platform that benefits business today and lays the foundation for grid services capabilities.

Instill Confidence with Security-Rich Features

WebSphere Application Server offers a sophisticated security-rich infrastructure, single sign-on capabilities and extensive support of open, standards-based Java and Web services specifications. WebSphere Application Server provides a secure infrastructure to help prevent unauthorized access to the J2EE technology and Web resources it protects, through strong authentication capabilities that consist of basic (userID/password), forms-based certificate and encryption options for secure user validation and roles-based authorization.

WebSphere Application Server Version, 6.1 addresses security concerns with default out-of-the-box security configurations protecting implementations even before custom configurations specific to environments are implemented. Also included is a default user registry to provide effective identity management with new installations. The WebSphere Application Server, Version 6.1 security infrastructure, through integral support and a pluggable security architecture, provides support for:

- Java Authentication and Authorization Service (JAAS), to authenticate new principles and manage privilege information for a principal.
- Java 2 Platform, Standard Edition (J2SE) V5.0 for securing system resources, and Java 2 Platform, Enterprise Edition (J2EE) V1.4 for securing and controlling access to WebSphere Application Server application and resources.
- Java Secure Socket Extension (JSSE), to help secure communications channels based on transport-level security (Transaction Layer Security [TLS] and Secure Socket Layers [SSL]).
- Java Cryptography Extension (JCE) framework for security encryption and message authentication.
- Java Cryptography Architecture (JCA) for Public Key Infrastructure (PKI) integration.
- Common Secure Interoperability V2 (CSlv2) for vendor-neutral, secure interoperability between application servers. This protocol adds additional security features that enable interoperable authentication, delegation and privileges in a CORBA environment. CSLv2 supports interoperability with

the EJB 2.1 specification and is the mechanism employed by WebSphere for interoperability between Java clients and EJBs, or between EJBs and remote ORBs.

- Pluggable user registries to enable the use of Lightweight Directory Access Protocol (LDAP) or custom registries.
- Simplified Certificate/Key management
- Web single sign-on provided exclusively with WebSphere software or through integration with front-end authentication end points with trust association interceptor (TAI) technology.
- Highly secure access to enterprise information systems (EISs) through a pluggable principal- and credential- mapping facility.

WebSphere Application Server includes integral Single Sign-on (SSO) capability across multiple WebSphere Application Servers and Lotus Domino servers in the same domain, using Lightweight Third Party Authentication (LTPA), with capability to integrate LDAP or custom user registries. In environments that handle cross-platform authentication, WebSphere Application Server, Version 6.1 supports the Simple and Protected GSS-API Negotiation Mechanism (SPNEGO) for seamless single sign-on experiences on the Windows desktop. This extension to HTTP Negotiate enables authentication tokens to be passed from a browser-based client on the Windows platform to a server based on a different platform using HTTP headers. With SSO support, web users can authenticate once and not be prompted for credentials again when accessing WebSphere Application Server resources (e.g. HTML pages, JSP files, servlets, EJBs, etc.) and Lotus Domino resources such as documents in a Domino database.

You can also leverage WebSphere's pluggable security architecture (Java Contract for Containers (JACC) standards-based support) to implement sophisticated enterprise topologies and infrastructures including the use of third-party authorization providers for access decisions. This includes: pluggable user registries to enable you to exploit the local OS registry, LDAP or custom registries. Web single sign-on can be provided by integral WebSphere SSO capabilities, pluggable integration of credential mapping services for access to enterprise information systems, or through pluggable integration with IBM or third party front-end authentication endpoints through Trust Association Interceptor (TAI) technology, with support for both TAI and TAI++ interfaces. For example, via TAI you could use a reverse proxy security server such as Tivoli WebSEAL to shield a Web server from unauthorized requests, combined with Tivoli Access Manager to provide a unified identity management and security solution.

For a centralized approach to security, WebSphere Application Server provides tight integration with the IBM Tivoli Access Manager. Integration with IBM Tivoli Access Manager provides a centralized identity management solution with global sign-on capabilities and enforceable policies to secure cached and noncached J2EE, portal, Web and legacy resources. The integration of Tivoli Access Manager with the WebSphere Application Server platform provides enterprise-wide management of authentication and authorization with SSO. Companies that implement this integrated solution can benefit from the ease of working with a single object namespace, representing the full set of security policies for the resources you want to protect, with greatly simplified administration and enhanced end user experience.

WebSphere Application Server's pluggable architecture also increases the extensibility of Web services security. The implementation includes many of the features that are described in the Organization for the

Advancement of Structured Information Standards (OASIS) Web Services (WS) Security Version 1 standard. In addition to support for SOAP Message Security (WS-Security) – which defines the propagation of security credentials, including identity assertions and digital signature support (UsernameToken 1.0, X.509 Certificate Token 1.0) – and XML-based encryption for more secure Web services interactions, WebSphere Application Server provides programming model extensions to support higher-level specifications proposed to extend the WS-Security specification. Examples of these specifications include WS-SecurityKerberos, WS-SecureConversation and WS-Trust. New in WebSphere Application Server, Version 6.1 is support for the WS-I Basic Security Profile 1.0, which promotes interoperability by addressing the most common problems encountered from implementation experiences to date.

Additional Web services security is provided by the Web services gateway component of WebSphere Application Server Network Deployment. This allows businesses to securely externalize and expose applications as Web services. For most secure consumption of web services (particularly those external to your organization), WebSphere Application Server, Version 6.1 supports the latest Universal Description, Discovery, and Integration (UDDI) V3, with its improved security features, including digital signing of UDDI entries.

To provide you with the assurance that WebSphere Application Server, Version 6.1 will satisfy your security objectives and that the security functions are correctly implemented, it has been designed to maintain its Common Criteria Assurance Level 4 certification. Common Criteria is a scheme for independent assessment, analysis, and testing of IT products to a set of security requirements. Certification gives customers the confidence that products will be effective in delivering functions such as identification and authentication, and user data protection.

Always On, Always Available

WebSphere Application Server Network Deployment, Version 6.1 is the foundation for high availability solutions that leverage world class clustering, edge-of-network services, distributed caching, improved workload management at all server tiers, and fault tolerance with the integral High Availability Manager (HAManager), and Backup Cluster support. In clustered configurations, WebSphere Application Server Network Deployment can deliver high availability and failover capability at the HTTP server, Web container, and EJB container application tiers.

At the HTTP tier, the Load Balancer, which is part of the Edge Components delivered in WebSphere Application Server Network Deployment V6.1, can provide both scalability and high availability for the web servers by distributing the incoming workload across multiple web servers. The Load Balancer is an edge-of-network system that directs network traffic flow, reducing congestion and balancing the load directed to back end services and systems. The Load Balancer provides site selection, workload management, session affinity, and transparent failover. In most cases it can also be configured to provide content-based routing. The Load balancer monitors the availability of web servers in the cluster and dispatches incoming requests to the appropriate server in the cluster based on both weighting metrics and availability. Because the Load Balancer is the entry point for clients accessing your application, high availability for this component itself is important. High availability for the Load Balancer can be achieved by setting up a replicate Load Balancer, supporting either an active/active configuration with mutual takeover on failure or an active/passive failover

configuration. Load balancing can be tuned for specific application and platform criteria using custom advisors code that feeds performance monitoring metrics into the load balancing manager for optimum server allocation.

The Proxy Server for WebSphere Application Server Network Deployment, Version 6.1 provides a high performing web intermediary that can dynamically route traffic to WebSphere applications with minimal configuration or administrative actions. It provides integrated workload management, enhanced dynamic caching support, improved clustering support and administration capabilities and can be used outside a typical Application Server environment to include routing and caching for any HTTP server as well. It includes support for IPV6 and 64-bit platforms.

For the Web container tier, the WebSphere web server plug-in provides workload management and failover across web containers in the cluster and can also distribute requests around (bypass) cluster members that are not available.

The Application Server provides distributed in-memory session replication and failover capabilities by leveraging the HAManager. Session state can also be preserved using database persistence.

Finally, the EJB tier achieves high availability through the combined functions of the EJB Workload Manager (WLM), across an EJB server cluster, and the HAManager, which protects the EJB WLM service with replication and failover support (eliminating a previous single point of failure).

The WebSphere Application Server Network Deployment High Availability Manager enhances availability by protecting critical singleton services (such as transaction and messaging services, or the EJB WLM service) from becoming single points of failure. The HAManager is responsible for running key services on available servers and supports hot standby and peer failover for these services. These automated hot failover capabilities deliver near real-time recovery when fail-over is detected in a cluster and permits, for example, peer recovery of in-flight transactions or messages among clustered WebSphere Application Servers.

Mirrored backup clustering in WebSphere Application Server Network Deployment, Version 6.1, can be used to failover EJB requests from a primary cluster to the mirror cluster if the primary cluster fails, with automatic failback when the primary cluster becomes available again. As such, without manual intervention, Backup Cluster Support will send a workload to another cluster if a cluster goes down.

WebSphere Application Server Network Deployment, Version 6.1 can take advantage of fault tolerant storage technologies such as Network Attached Storage (NAS). For example, WebSphere Application Server Network Deployment, Version 6.1 can be configured to store transaction logs for each server on a NAS shared file system. This allows all peers to see all the transaction logs, permitting rapid recovery of a transaction by any peer in the cluster, while lowering the cost and complexity of high availability configurations.

Continuous Availability and High Performance Computing

To help avoid costly system downtime, WebSphere Application Server Network Deployment Version 6.1 is designed to eliminate single points of failure using a high-availability framework that provides peer-to-peer failover for applications and processes running within a WebSphere environment. The built-in HAManager helps simplify the configuration of high-availability systems, and provides quicker failure detection along with

faster recovery times. You can also integrate WebSphere Application Server Network Deployment into an environment that might be using other high-availability frameworks, such as IBM High-Availability Cluster Multi-Processing (IBM HACMP) to manage non-WebSphere resources. And with near continuous uptime for critical applications, WebSphere Application Server Network Deployment, Version 6.1 can help you to ensure that business opportunities are not missed due to applications being unavailable.

For satisfying even the most extreme implementations, WebSphere Extended Deployment builds on the WebSphere Application Server Network Deployment foundation to achieve near-continuous availability for high volume business critical transactions, previously only achieved with traditional transaction processing monitors.

WebSphere Extended Deployment delivers dynamic operations enhancements which allow your application environment to scale on demand with improved resource utilization and lower TCO. With the WebSphere Extended Deployment Business Grid, resources are virtualized in a common pool so that they can be shared amongst multiple, transactional applications and long running workloads, including batch processing and computationally intense applications (such as scientific, CAD, statistical modeling, etc.). With the goals directed infrastructure, workloads are classified, prioritized, queued and routed according to customer business goals and relative application importance. Automatic managers provide dynamic adjustment of application resources based on actual demand and workload.

The On Demand Router (ODR) is a major functional element of WebSphere Extended Deployment that provides this business goals and service policy directed workload management. The ODR sits in front of the WebSphere Extended Deployment cluster and controls and shapes the traffic to the clustered servers. When an incoming request arrives, the ODR first classifies the requests based on business goals (e.g. a stock trading application could be assigned a higher business importance than a portfolio advice application) and places the incoming requests on a processing queue that corresponds to their service class. The ODR runs an Autonomic Request Flow Manager (ARFM) that prioritizes these requests, based on defined service policies (for example, to enforce a service level agreement), and controls the rate and order of traffic flow (e.g. stock trading vs. portfolio advice) to the routing and load balancing component which delivers the requests to the appropriate servers.

High Performance Computing facilities in WebSphere Extended Deployment extend the WebSphere Application Server Network Deployment environment to optimize high volume transaction processing environments, achieve near-continuous availability, and to provide near-linear scalability with commodity hardware as transaction load increases. WebSphere Extended Deployment addresses these capabilities with the Partitioning Facility, high end caching with the ObjectGrid, and by extending autonomic High Availability management to key application singleton services in addition to the application server HA management delivered by WebSphere Application Server Network Deployment server configuration. The ObjectGrid is an extensible, fault tolerant, transactional object caching fabric delivering quick and easy object data sharing and high performance access that improves application scalability and performance. For high volume OLTP applications (such as equity trading, reservation systems, etc.) the Partitioning Facility addresses the bottlenecks that can occur with intensive read and write operations. It enables the partitioning of applications, the ObjectGrid, and data (e.g. based on stock ticker symbol, etc.), improving database as

well as in-memory caching and workload management, dramatically decreasing contention for shared data and application resources. With clustered application partitions, WebSphere Extended Deployment can deliver low response times, near continuous availability, and near linear scalability on commodity hardware.

Enhanced User Experiences

For improved system response times and enhanced user experiences, you can deploy edge-of-network services using the Edge Components of WebSphere Application Server Network Deployment, Version 6.1. The Edge Components provide sophisticated load balancing, edge-of-network caching, and centralized security services to address the needs of high volume, highly available web application environments. In addition to the web server load balancing and failover capabilities (discussed above), the Edge Components Caching Proxy has the ability to cache, serve, and invalidate not just static HTML but also dynamic content in the form of JavaServer Pages and servlet responses generated by a WebSphere Application Server creating a virtual extension of the application server dynamic caches out into edge-of-network caches. Edge-of-network caching technology can reduce network congestion by storing frequently accessed content so information need be retrieved only once. Information can be cached and invalidated depending on when it will expire, how large the cache should be, when the information should be updated, or invalidated or refreshed in response to a back end application event. Edge caching can improve response time (and user experience) dramatically and offloads unnecessary processing from the backend servers, while maintaining tight integration with enterprise security and access managers (via plug in support) with Tivoli Access Manager, LDAP directories, or third party authentication and authorization mechanisms.

Caching can be further extended out into network based caches (such as the Akamai network) through use of Edge Side Include (ESI) technology. ESI mechanisms permit pages to be assembled from cached content (page fragments) at the edge of the network, thereby reducing bandwidth consumption and decreasing response time further. ESI is a simple markup language and proposed open standard (finished public review in the Java Community Process) for the dynamic assembly of Web page fragments, such as stock quotes and individual catalog prices. By leveraging ESI technology, dynamic content caching is extended by moving fragments from the application server to a proxy server that resides in the network, such as the Akamai network. This enables caching to occur at a more granular level and allows companies to position page composition at the most optimal location, closer to the end user. As a result, companies can improve user experiences through expedited, personalized page composition and help reduce workload on the network servers that occurs because of fragmented offload to the edge. This is particularly useful in a B2C scenario for delivering personalized pages to end users, whereby only the personalized page elements need come from the backend application and most elements could be served from static or dynamic content in edge-of-network or out-in-network caches.

Harness the Power of z/OS

WebSphere Application Server for z/OS, Version 6.1 brings the qualities-of-services of the z/OS platform to J2EE and Web services applications to support the demanding requirements of large-scale on demand enterprise computing. WebSphere Application Server for z/OS, Version 6.1 is specifically optimized to utilize the unique qualities of services provided by IBM System z™ hardware and the z/OS™ operating system, providing the ultimate in availability, scalability, and security. These unique capabilities translate into real, tangible business benefits for the demanding requirements of large scale on demand enterprises.

- *Eliminate Lost Business Opportunities:* WebSphere Application Server for z/OS, Version 6.1 provides an on demand infrastructure for near continuous uptime for mission critical applications. z/OS is capable of consistently delivering expected service regardless of capacity-constrained environments, unanticipated workload spikes, or failures in applications, system software or hardware. WebSphere Application Server for z/OS features the High Availability Manager (HA Manager) to assist in monitoring and recovering WebSphere servers, resources, and components. This works in synergy with the underlying WebSphere Application Server for z/OS design which focuses on workload isolation; exploitation of z/OS Parallel Sysplex clustering; integration with z/OS Automatic Restart Management (ARM) and utilization of the self-managing behavior of the z/OS Workload Manager (WLM).
- *Service Level Agreements:* WebSphere Application Server for z/OS leverages the z/OS WLM to provide performance goal-oriented workload balancing, management and reporting within a system and across a Parallel Sysplex cluster. The eWLM integrates with the native WLM capabilities of z/OS to extend reporting and monitoring of transaction performance, across conformant platforms of the multiple, disparate servers that are invariably involved in web-centric transactions. WebSphere Application Server for z/OS enables the ability to help ensure service levels (response time, throughput and so on) for specific types of customers and high-priority workloads as defined by business needs.
- *Integration and Extension of Existing Assets:* Composition and integration with multiple z/OS resource managers is a key requirement for any application that needs to reuse existing assets. WebSphere Application Server for z/OS is designed to provide optimized, heterogeneous two-phase commit concurrency control with IBM IMS™, CICS and DB2. Using WebSphere Application Server for z/OS as your integration engine can provide optimal performance (through closer data proximity and a reduced duration of held locks), better availability (through reduced points of failure) and faster recovery in rollback situations. For example, unique connections are available to DB2 UDB for z/OS that simplify the architecture and enhance your application performance.
- *Lower TCO with dedicated processors:* WebSphere Application Server for z/OS Version 6.1 supports the System z Application Assist Processor (zAAP) which can drastically reduce the cost of running new Java workloads on the mainframe. Available on z890 and z990 (and follow on models), the zAAP is a dedicated processor for Java workloads that can operate asynchronously with the general purpose processors, free from ISV or OS processor charging. This allows you to offload processing cycles from the general purpose processor, improving performance and reducing TCO (via freeing up cycle charges from the GP processor). Similarly, the new System z9 Integrated Information Processor (zIIP) helps

offload DB2³ processing much like how the zAAP offloads Java processing from the general central processors.

- *Efficiency:* Maximizing people and system resources. z/OS is designed for efficiency and can provide a lower total cost of ownership through reduction in trained system programmers to configure, monitor and adjust multiple systems, and fuller utilization of existing capacity. z/OS is able to automatically handle unpredictable spikes in mission-critical workload without wasting spare cycles during periods of low and average utilization. WebSphere Application Server for z/OS easily fits into the heterogeneous nature of z/OS workloads running simultaneously in either a single z/OS image or across multiple images configured in a Parallel Sysplex cluster.
- *Security:* z/OS is a proven security leader, maintaining the integrity and availability of systems, applications and data in the face of threats. WebSphere Application Server for z/OS is designed to fully integrate with the IBM Tivoli Access Manager, z/OS Security Server for z/OS or equivalent z/OS security products.

For the existing z/OS customer, WebSphere Application Server for z/OS helps you fully leverage your existing investment in System z in new ways. Utilize the skills and procedures already in place for your data center with WebSphere Application Server for z/OS as you extend and reuse these battle-tested, proven assets in web, traditional J2EE or SOA environments.

WebSphere Application Server for z/OS provides the best of both worlds for your environment—the deep exploitation of the System z hardware and z/OS software, with the application portability of J2EE 1.4/J2SE 5.0 and Web services standards. As business needs demand, you can redeploy a J2EE application or Web service already deployed to another platform without code changes to z/OS. With WebSphere Application Server for z/OS, you can leverage your existing assets and investments without new skills or hardware purchases, and you can reuse well-established operational procedures for System z to provide a strong foundation for your SOA environment.

³ DB2 for z/OS V8 will be the first application to support the zIIP.

Extensive Communication Services

More and more, customers are searching for ways to increase the flexibility of their business processes, streamline their IT infrastructures, and retain and reuse their existing assets. These customers are looking to standardize, automate, and integrate their business processes to allow greater flexibility in adapting to dynamic business conditions for competitive advantage. Customers need implementation technologies that are standardized, interoperable, and support the composition of applications from loosely coupled, often asynchronous and reusable services. At the same time, most customers have existing IT assets and investments in enterprise information systems that they need to extend and reuse within a SOA environment. IBM WebSphere software provides a strategic foundation for on demand solutions through a comprehensive platform that is implemented as a SOA. IBM WebSphere Application Server, Version 6.1 is the key building block for SOA applications, with leading support for standardized implementation technologies that allow you to build SOA applications, including support for the latest J2EE 1.4, J2SE 5.0, and Web services standards, and a unified communications infrastructure that allows loosely coupled, reliable, asynchronous interactions among J2EE components, Web services and legacy systems.

Web Services Standards

Web services are self-contained, modular applications that can be described, published, located, and invoked over a network. They are the key standardized technology for implementing application functionality as reusable services. They implement a service oriented architecture, which supports the connecting or sharing of resources and data in a very flexible and standardized manner. As such, Web services provide a standardized way of encapsulating business processes in new and existing applications, and exposing them for reuse and integration within a SOA. Because WebSphere Application Server features an open approach to transforming any application asset into a modular service, you can have these services accessible for reuse by other developers throughout your organization. Web services-based software aids internal development and integration, and it provides a standardized method for publishing encapsulated business services to drive business-to-business (B2B) collaboration and, in time, new revenue streams. Examples of these services include stock quotes and charting, credit card verification and payment processing, integrated travel planning and auctioning.

WebSphere Application Server, Version 6.1, expands on its support for Web Services (WS) standards incorporating latest enhancements to provide better application portability and control, performance improvements, and business transaction coordination in SOA implementations. For example, WebSphere Application Server Version 6.1 has added support for WS-Notification, WS-Business Activity, WS-Interoperability Basic Security Profile, SOAP/JMS enhancements, and changes to SOAP with Attachments API for Java (SAAJ).

The WS-Notification (WS-N) specification provides an event-driven, or publish-subscribe programming model for Web Services applications, which helps standardize the way Web services interact, and provides better interoperability between Java and non-Java environments. Similarly, the WS-Business Activity (WS-BA) specification defines protocols to standardize the way transactions across different vendor implementations are rolled back in a multi-phase commit process. Where long running processes prohibit

the locking of data resources to make actions tentative, WS-BA allows applications, as opposed to databases, to control rollbacks. This allows applications spanning trust boundaries to support two-phase transactions while, if required, still providing a way for applications to logically undo any work deemed to have failed subsequent processing. For example, a process that books an entire vacation package, may book a flight, car rental, and hotel from three separate vendors. If a suitable hotel cannot be booked, the flight booking and car rental can be undone without incurring penalties for cancellation.

Session Initiation Protocol (SIP)

Session Initiation Protocol is a peer-to-peer protocol that was developed by the IETF MMUSIC Working Group and is a proposed standard for initiating, modifying, and terminating an interactive user session that involves multimedia elements such as video, voice, instant messaging, IP Television (IPTV), online games, and virtual reality. While SIP is the leading signaling protocol for Voice over IP and is primarily used in setting up and tearing down voice or voice calls, it can be used in any application where session initiation is a requirement (such as Event Subscription and Notification, Terminal Mobility, etc.). The JSR 116 specification enables the implementation of SIP using the Java Servlet model. For a new communications experience, WebSphere Application Server, Version 6.1, provides the facilities for Session Initiation Protocol (SIP) servlets as a core part of the application server.

WebSphere Application Server Version 6.1 also provides support for Converged Servlet Containers. These containers provide the facilities to share application sessions between varying servlets that utilize disparate protocols (such as HTTP servlets and SIP servlets), to allow interaction with one another. The combination of SIP and Converged Servlet Containers provides the facilities to create powerful applications that incorporate video, voice, and instant messaging capabilities, of which the state of these activities can be visible to the originating application. For example, call center staff could initiate customer service calls directly through CRM applications which are aware of state information allowing the recording of accurate call durations for future business analysis, while simultaneously alleviating the requirement for staff to input manual entries.

SIP tooling has been added to the Application Server Toolkit to help you more easily build SIP applications. Additionally, WebSphere Application Server Version 6.1 enhances its Proxy Server to provide SIP Servlet clustering capabilities and the necessary availability and scalability expected for today's business applications.

Integration of Asynchronous Messaging Services

In a SOA, application functions (business processes) are modular and inherently loosely coupled. SOA applications need to initiate and respond to asynchronous invocations, conversations and broadcasts. Because of this, SOA applications and integrations require a standardized communications infrastructure that allows loosely coupled, reliable, asynchronous interactions among J2EE components, Web services, and legacy systems capable of messaging. While WebSphere Application Server itself provides support for mixed synchronous and asynchronous transactional environments, the messaging resources of WebSphere Application Server, Version 6.1, seamlessly interoperate with IBM WebSphere MQ, allowing you to combine these products to form the basis of a powerful service bus that can integrate the most diverse applications and environments.

WebSphere Application Server, Version 6.1, provides a built in Java Message Service (JMS) provider, fully integrated within the application server. This provides a full implementation of the JMS 1.1 standard including support for the publish and subscribe model, and a full range of messaging Quality of Service options from best-effort to assured-persistent messages. Data objects can be persisted to a choice of a data store, the embedded IBM Cloudscape™ Java database or an external database resource provider (such as IBM DB2 or other vendor's database), or to a file system store providing improved performance and easier configuration and management. Being fully integrated, the native JMS provider can scale with the application server (clustering) and take advantage of the high availability features of the platform (e.g. failover and HAManager protection for the JMS engine). In WebSphere Application Server Version 6.1 you can now add a WebSphere MQ for z/OS queue manager or queue sharing group as a member of a service integration bus, thereby providing more integration between IBM WebSphere MQ for z/OS and the default messaging provider in WebSphere Application Server. This will help improve availability and simplify administration.

WebSphere Application Server, Version 6.1, also incorporates a host of additional messaging related improvements including, but not limited to, JMS performance improvements with changes to SOAP/JMS and changes to SOAP messages passed to and from Web Services using Java. SOAP with Attachments API for Java (SAAJ) helps to automate many of the programming steps required to create SOAP connections and messages, to populate and send messages, and to receive replies. Based on the Java API for XML messaging, SAAJ can result in performance and productivity enhancements over the manual generation of messages.

Service Integration Bus Architecture

A unified communications infrastructure for connecting services is a key element of SOA initiatives. A service integration bus (SiBus) is a logical concept that provides an intelligent infrastructure for integration and connection of services in a SOA. Defined within a WebSphere Application Server cell, a SiBus is composed of the messaging engines, a component of the application server created when a server is associated with the SiBus, and related services of the WebSphere Application Servers collaborating together along with integrated Web services functionality (Figure 6). The SiBus provides a common infrastructure for secure communications between application services through synchronous and asynchronous messaging. As such, the SiBus provides a highly available and scalable, unified communications pipeline that enables application services to participate transparently in message exchanges, whether the services end points are: Web services providers and requestors, J2EE Connector Architecture (JCA) resource adapters and application servers JCA containers, or JMS messages. The SiBus provides additional features needed to connect the components of a SOA architecture, including protocol translation between HTTP Web services and JMS Web services. The SiBus can also connect to WebSphere MQ, participating in an extended bus infrastructure, or with other instances of a SiBus in the same or other WebSphere Application Server cells, for enterprise wide SOA applications and integrations. The interconnected network of SiBus instances and WebSphere MQ queue managers provides the underpinning messaging infrastructure upon which an Enterprise Service Bus can be built by introducing products such as WebSphere Enterprise Service Bus and WebSphere Message Broker to mediate (message transformation, content based routing, logging, etc.) between applications and provide advanced protocol transformations.

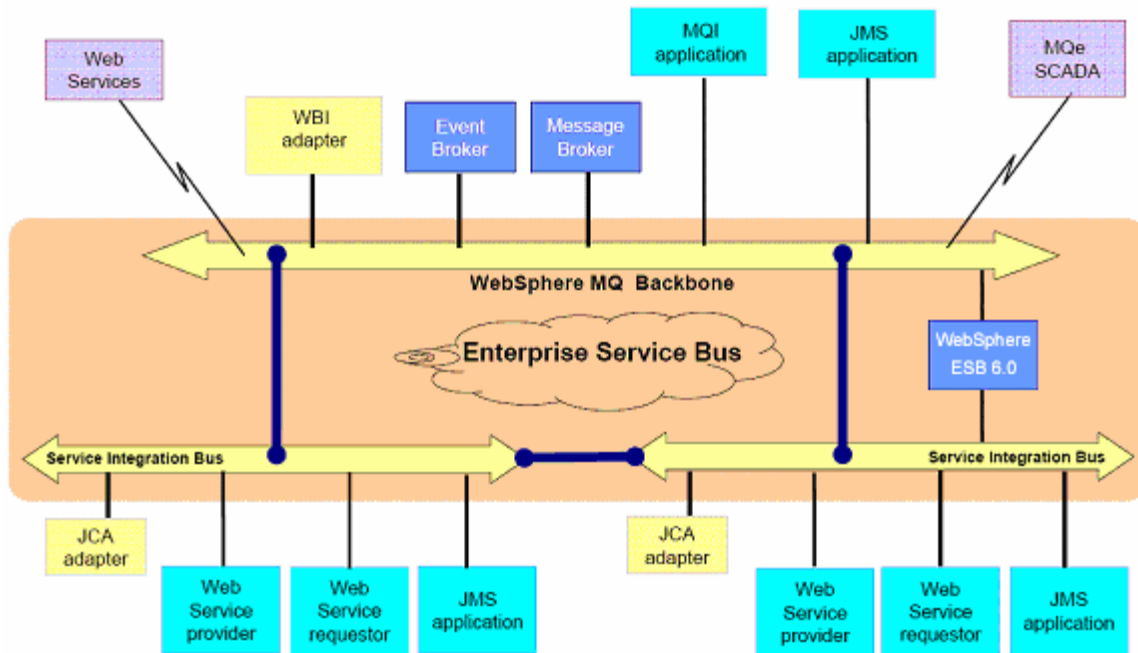


Figure 6 - Service Integration as part of the Enterprise Services Bus

With WebSphere Application Server, Version 6.1 you have a comprehensive platform for enterprise SOA. With an open service-oriented architecture and a unified communications bus infrastructure, you can easily integrate new and existing assets, increase your business flexibility and facilitate the efficient use of your IT resources.

Integrate and Reuse Existing IT Assets

Many customers have significant investments in legacy systems and enterprise applications. Customers need to extend and reuse these existing assets in conjunction with new business logic written in Java to maximize value and achieve secure, optimum resource utilization. The ability to create new applications that incorporate a variety of enterprise resources quickly and easily is a key requirement in a variety of industries. Businesses are looking for new ways to reduce the complex underlying coding required to create these dynamic applications without sacrificing transactional integrity. For example, an insurance or telecommunications company that stores a multitude of customer data utilizing IBM CICS® now needs to integrate this information within its J2EE environment.

WebSphere Application Server helps reduce the risk, complexity and cost of extending enterprise applications through its advanced support for J2EE Connector Architecture (JCA) 1.5. The JCA specification defines a standard architecture for integration between heterogeneous enterprise information systems (EIS) and application servers such as WebSphere. This gives you a consistent way to connect to and communicate with a wide range of enterprise systems and applications as well as advanced transaction coordination, without the need for advanced programming skills or extensive coding. Based on the JCA specification, an EIS vendor (e.g. SAP, Oracle / PeopleSoft, Siebel, etc.) or third party connector developer can develop a standard resource adapter for its EIS to plug into any application sever that supports JCA.

JCA 1.5 supports both synchronous and bi-directional, asynchronous interactions between a J2EE server and resource adapters as well as JMS provider support. Via JCA support, WebSphere provides client applications all system services regarding connection, transaction, and security management on behalf of the resource managers. Comprehensive J2EE Connector Architecture support is also provided at development time (in Rational Application Developer) with tools that let a developer easily integrate existing EIS such as CICS or IMS™ into their SOA solution. Using these tools, developers can quickly create a Web service from an existing EIS transaction.

JCA has two basic components:

- A common client interface (CCI) that manages the flow of data between the application and the back-end system and has no visibility into how the container and application server perform
- A set of system-specific services implemented as part of its base J2EE platform

CCI is a programming interface that application developers and client programs can use to connect and access back-end systems. It is a low-level application programmer interface (API) and is similar to Java Database Connectivity (JDBC). Unlike JDBC, however, CCI can work with non-relational systems. Although it's possible for application developers to call CCI directly, in most cases an application developer will write to an abstraction layer, provided by the connector provider or enterprise application integration (EAI) framework vendor, simplifying the development process. To simplify development, WebSphere Application Server, Version 6.1 provides a relational resource adapter (RRA) that has an implementation for both the CCI and the traditional JDBC interfaces.

On the platform side, JCA defines a set of service contracts that a connector developer can expect will be available to the adapter at application runtime. The primary services defined in JCA 1.5 and implemented in WebSphere Application Server include:

- *Connection management* - Enables WebSphere Application Server to create and manage connections to back-end systems. WebSphere Application Server also implements connection pooling, because connections to back-end systems are expensive.
- *Transaction management* - Supports XAResource transactional access to underlying resource managers. This service enables the transaction manager within the EJB server to manage transactions across multiple back-end systems. Transaction Inflow is a new feature of JCA 1.5, which allows a resource adapter to propagate an existing two phase transaction into the application server.
- *Security management* - Enables the developer to define security between the EJB server and the back-end system. The specific security mechanism that is used is dependent on the security mechanism provided by the back-end system.
- *Work-management* - Supports synchronous or asynchronous execution of work with timed, delayed, or periodic work execution central to transactional control in long running workflows.
- *Message-inflow* - Enables simplified and expanded support for message driven beans. A message driven bean is an enterprise bean that allows J2EE applications to process messages asynchronously, unlike session or entity beans that allow you to send JMS messages and to receive them synchronously only. The message driven bean acts as a JMS message listener and contains methods for message processing. This provides ability for a resource adapter to invoke an application asynchronously through

a message-driven bean. Essentially this allows plug-in of a variety of messaging providers into an application server.

With WebSphere Application Server, Version 6.1, IBM delivers a robust infrastructure for SOA. WebSphere Application Server, Version 6.1 enables customers to create a SOA that is the path to business flexibility and on demand business. WebSphere Application Server, Version 6.1 helps customers to increase their return on existing investments and lower total cost of ownership with support for the latest web services standards, a unified communications bus infrastructure, standards-based integration of existing applications, and optimal resource utilization in a secure and robust transactional environment.

Agile Administration and Management

WebSphere Application Server systems management has been significantly enhanced in Version 6.1. It is now easier to deploy, manage, and tune the J2EE and Web services composite applications that are typical of SOA environments. WebSphere Application Server, Version 6.1 has a unified deployment and administration model that spans all configurations, from single servers to highly distributed, clustered, high availability configurations. This unified administration facility across the WebSphere Application Server, Version 6.1 configurations, minimizes training costs and makes it much easier to migrate across the Version 6.1 family – from Express to Network Deployment and z/OS.

Efficient Deployment

WebSphere Application Server, Version 6.1 provides improved flexibility to deploy your applications across the various configurations of the platform. Through the introduction of an administrative console wizard, you can update deployed applications or modules by: replacing the entire application (EAR file); replacing, adding, or removing a single module (WAR, EJB, JAR, or connector RAR file); replacing, adding, or removing a single file; and replacing, adding and/or removing multiple files by uploading a compressed file. If applications are running when updated, WebSphere Application Server, Version 6.1 will automatically stop the applications or only the affected components, update the application logic, and then restart the stopped application components. Previous versions of the WebSphere Application Server only supported the replacement of an entire application and always stopped and restarted the entire application for any change.

In multi-server and clustered topologies, WebSphere Application Server Network Deployment makes deploying and updating applications similarly seamless, providing centralized administration through its Deployment Manager. Deployment across clusters allows you to install, update, or delete an application and have the updates automatically distributed to all members in a cluster. Incremental updates are supported, allowing a cluster cell to be upgraded to a new release one node at a time, which minimizes the impact to applications running within the cell, improving availability. In the previous version (WebSphere Application Server V5), if you updated an application on a cluster, you had to stop the application on every server in the cluster, install the update, and then restart the server. With WebSphere Application Server, Version 6.1, the rollout update option allows you to update and restart the application servers on each node, one node at a time, and providing continuous availability of the application. You can add more WebSphere Application Server nodes dynamically and manage them in the network asynchronously, regardless of individual or overall network status and with updates assured when server and network communication is available.

For clustered implementations, WebSphere Application Server, Version 6.1 cluster creation includes a number of initiatives to improve the creation and configuration of clusters. Specific improvements include:

- wizard enhancements to aid in cluster creation;
- panel re-factoring to show available cluster members in the improved cluster members properties panel and not the “application server” panel, avoiding any confusion over servers that are members of a cluster versus those that are not; and
- the provision of guided activities to aid in deploying applications in a continuously available environment.

Additionally, cluster weight, which represents the distributed weight that is allocated among servers in a cluster, has been enhanced to direct editing of the configured cluster weight and the actual runtime weight. Cluster weights are displayed in a table to show the relative weighting compared to other items in the table.

WebSphere Application Server, Version 6.1 also provides improvements for troubleshooting and addressing any potential issues with your implementation should they arise. In addition to the help and guidance available through an extensive community of users and partners, WebSphere Application Server, Version 6.1 now includes the IBM Support Assistant. The Support Assistant is a free stand-alone application that is now included with WebSphere Application Server to save you time by bringing many of IBM's support resources to you in a single application which can be enhanced through plug-in modules for all the IBM products you use. The Search component searches multiple IBM and non-IBM locations to lead you to the most pertinent support information providing quick access to resources such as product pages, support pages and news groups. Where IBM service requests are required, the Support Assistant eases the collection of system information data needed to analyze software problems, creating and tracking your electronic problem report. Additionally, new Diagnostic Providers expose information about running components, enabling administrators to more easily debug problems related to the components. You can detect problems faster and access more information for determining and solving problems.

Effective Application Management

IBM WebSphere Application Server, Version 6.1 provides administrators with a web-based, graphical Administration Console which is based on the Integrated Solutions Console (ISC). ISC provides a framework for administering multiple products so administrative functions are handled through portlets (using the JSR-168 standard), or components, within a single system. Each product provides product-specific functionality within a plug-in to the Integrated Solutions Console framework, and an Eclipse plug-in allows the development of add-on console components. This central administration portal provides easy access to all aspects of server, cluster, dynamic cache, and resource management, including resources participating in the Service Integration Bus architecture (such as the integrated JMS services and Web Services Gateway). This unified administration console offers consistency across all WebSphere Application Server, Version 6.1 configurations and beyond, which contributes to reduced training costs and lower total cost of ownership overall.

Additional features of this Administration Console include:

- Integration of the Deployment Manager (WebSphere Application Server Network Deployment) into the administration console providing a single administrative interface for managing a logical group of servers across clusters of nodes, cells, and machines.
- Unified management for mixed deployment environments of WebSphere Application Server V5.0, V5.1, V6, and Version 6.1 for Network Deployment and z/OS configurations.
- Adapt-a-View support that tailors the console view based on the administrator's role (privileges), and the version, platform, and installed capabilities of your WebSphere Application Server environment.
- Integrated management for the IBM HTTP Server which is now simplified through easier security configurations, an enhanced cluster creation wizard, and simplified Database connectivity.

- Enhanced performance monitoring tools, including performance monitoring of dynamic caches of the application server and Edge components in the Network Deployment configuration, and Tivoli Performance Viewer integrated into the console (Figure 7) for all Version 6.1 configurations.

IBM is also including a new standalone thin administration client with WebSphere Application Server, Version 6.1 to allow you to install an administrative client and associated scripts on remote machines with limited disk capacity. This client can be packaged as a JAR file or as an Open Services Gateway Initiative (OSGi) based bundle. The OSGi specifications define a standardized, component oriented environment for networked services that provide the capability to manage the life cycle of software components from anywhere in the network. Since customer applications load only those application server components required, memory footprint requirements can be significantly reduced.

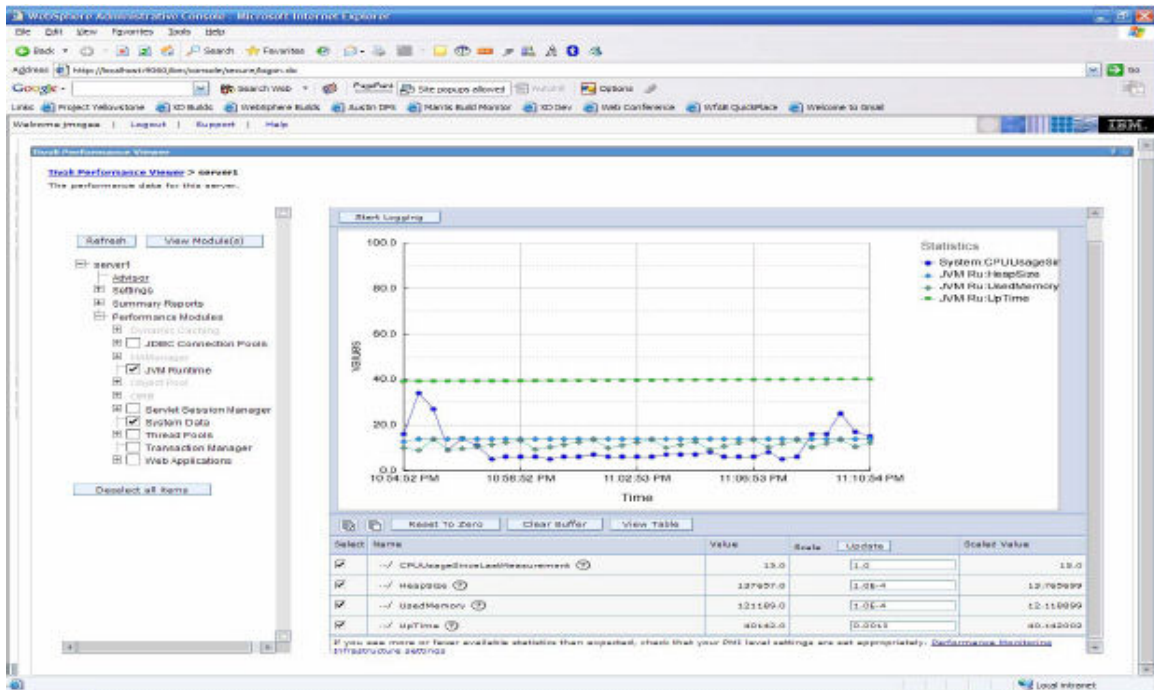


Figure 7 - Tivoli Performance Viewer integrated in Administration Console

In addition to the Administration Console, WebSphere Application Server supports traditional scripting administration tools (such as the wsadmin, ANT, and Jython files). As an alternative to the drag and drop capability for modeling scripts introduced in Version 6.0, a new Console Command Assistant is provided in the administrative console to save you time and money by automatically mapping your administrative activities to wsadmin scripting commands, so that you can quickly capture your console knowledge and apply it to wsadmin.

The WebSphere administration model is based on Java Management Extensions (JMX), and as such is highly extensible, allowing the integration of external administration solutions via the JMX API. JMX is a framework that provides a standard way of exposing Java resources, for example application servers, to a system management infrastructure. Within the standard WebSphere infrastructure, all of the managed resources are represented as JMX MBeans (JMX MBeans represent the management interface for a

particular piece of logic). This provides you with an open and highly extensible administration infrastructure that can be extended with MBeans to include other administration applications. JMX support also allows you to use external enterprise solutions (such as Tivoli security products) to manage WebSphere software in a standard way. New in WebSphere Application Server, Version 6.1, you can develop and build your own Java Management Extensions (JMX) client program that is compliant with the JMX Remote application programming interface (JSR 160). JSR 160 defines how JMX clients communicate with JMX servers. After you have a working JMX client program, you can use it to manage WebSphere Application Server or non-WebSphere Application Server systems allowing ISVs to create a single custom management interface for entire infrastructure deployment.

With WebSphere Application Server Network Deployment, base administration capabilities are extended to help manage configurations that include large numbers of servers. Automated application-server management functions help enhance productivity and help reduce administrative costs. The roles-based, tiered administration system and automated management functions dramatically simplify tasks, with automatic distributed process discovery (e.g. when a server starts and joins a cluster), and automatic publishing and synchronization of configuration information between the master configuration repository and each cluster node. In a WebSphere Application Server Network Deployment environment, multiple WebSphere Application Server nodes are managed from a single central location. However, administration tasks and configuration files are distributed among the cluster nodes and each application server starts from its local configuration file which reduces the reliance on a central repository or administration server. Although a master configuration is maintained via the Deployment Manager, this is pushed out (published) and automatically synchronized to the local node configurations, providing a unified view and automated approach to cluster wide management.

Extended Manageability with WebSphere Extended Deployment

To reduce the complexity and cost of managing complex distributed environments, WebSphere Extended Deployment delivers extended manageability features on top of the core capabilities provided with the Network Deployment configuration. Extended manageability facilities offer simpler and improved management of complex system operations with real-time advanced, meaningful visualization tools, controlled implementation of autonomic capabilities, and application edition management. These key features help to reduce the complexity and cost of managing your IT resources in highly distributed and clustered environments.

Visualization Tools provide “at a glance” understanding of your systems, including the status and health of applications running in the WebSphere Extended Deployment dynamic shared resource environment in the context of how they are meeting your established business goals. WebSphere Extended Deployment extends the Administration Console with three primary visualizations (Runtime Topology, TreeMap, and Custom Charting) that allow rapid understanding of your application deployments in the dynamic, shared resource environment of WebSphere Extended Deployment. These visualizations are updated in near real time to reflect the on demand changes to the environment, allowing full understanding of autonomic management capabilities of the dynamic WebSphere Extended Deployment environment.

Autonomic capabilities in WebSphere Extended Deployment help you to reduce the cost (and complexity) of managing your IT resources. Autonomics deliver increased system resiliency, responsiveness, and availability, with autonomics covering:

- WebSphere Extended Deployment goals directed WLM
- Health Policies for early proactive detection and automated correction of system problems
- Self managing and self allocating server resources such as dynamic clusters
- High availability management for server and application services with the HAManager.

Importantly, WebSphere Extended Deployment provides a gradual controlled implementation of autonomic capabilities, with manual, supervised, and on demand modes. This allows you to maintain control and incrementally adopt and leverage WebSphere Extended Deployment autonomic computing capabilities.

Application Edition Management (AEM) in WebSphere Extended Deployment simplifies the process of rolling out updates to your business applications without interrupting service and while maintaining high availability. Application Edition Manager allows you to have multiple application editions deployed to a WebSphere cell. The Application Edition Manager makes it possible to choose and control which edition to activate on a WebSphere Application Server cluster, enabling you to either rollout new application updates, or revert to previous editions as required, while maintaining availability. Application Edition Manager supports rolling upgrades and atomic upgrade strategies for rolling out application editions. The Application Edition Manager also supports a Validation deployment mode which is tremendously useful during testing to verify functionality of a new application edition in your production environment without exposing it to your real users.

Performance Monitoring

WebSphere Application Server Version 6.1 provides performance improvements at the individual server level resulting in overall performance enhancements - whether implementations are targeted at large scale clustered environments or individual servers. For example, WebSphere Application Server, Version 6.1 introduces support for J2SE 5.0 and the IBM Developer Kit (SDK) for Java 5.0 Platform. This introduction provides multiple performance enhancements including the asynchronous compilation of J2SE 5.0 Java methods to improve startup time of applications with a high level of multithreading on SMP servers, enhancements to garbage collection to help reduce the number of long pauses in the JVM runtime environment, and shared classes. In server environments where multiple JVMs exist on the box, shared classes eliminates the need to load all classes for each JVM instance. The use of a shared memory cache also results in faster and reduced memory usage. Similarly, enhanced dynamic caching support and improved clustering support in the proxy server used within a WebSphere Application Server implementation are designed to improve performance, and hence scalability.

WebSphere Application Server provides a comprehensive performance monitoring infrastructure and tools which allow administrators to easily visualize and analyze performance data collected in real time from various areas of your business application environment. This allows you to make informed decisions when it comes to diagnosing and performance-tuning the environment. WebSphere Application Server implements the J2EE 1.4 standard Performance Data Framework, with application server elements instrumented and exposed using JMX (JMX interoperability is supported between WebSphere Application Server, Version 6.1

and back level nodes at Version 5.x or at Version 6.0.2 and later). WebSphere Application Server provides a comprehensive performance monitoring solution out-of-the-box, with an integral Performance Monitoring Infrastructure (PMI), integrated performance visualization and analysis tools, and intelligent Performance Advisors.

- Performance Monitoring Infrastructure collects performance metrics data from the running application server and makes them available via JMX. The PMI collects data on your application resources (e.g. EJBs, servlets/JSPs, Web services, JMS, SiBus, or custom PMI), the application server's runtime resources (e.g. JVM memory, thread pools, database connection pools, dynamic caches, etc.), and system resources (e.g. processor usage, total free memory, etc.). In WebSphere Application Server Version 6.1, the Performance Monitoring Infrastructure now provides greater granularity with new counters for per-process CPU data, new transactional data for J2EE Connectors, and new Request Metrics (allows you to trace response times for individual transactions) for J2EE Connector, JDBC, EJB, servlet, and JMS resources.
- Tivoli Performance Viewer (TPV), integrated into the Administration Console (Figure 7 above), allows you to visualize the collected PMI data from local and remote application servers in your environment. TPV provides summary reports that show key areas of contention (e.g. EJBs and their methods, thread pools, connection pools, etc.), graphical and tabular views of raw PMI data allowing you to drill down to view individual performance metrics, and give you the ability to save PMI data to logs for detailed review.
- Performance Advisors are integrated into the Administration Console and Tivoli Performance Viewer. They provide tuning advice formulated from the gathered PMI data and configuration data. This determination requires considerable expert knowledge about the various components in the application server and their performance characteristics. The Performance Advisors generate intelligent tuning advice in real-time or offline, jump starting the tuning process and lowering the "expert skills" bar that otherwise would be required.

The Performance Monitoring Infrastructure in WebSphere Application Server is extremely extensible because it is based on open, industry standards and PMI data are exposed in a standards based manner (JMX). Support for custom PMIs allows you to add your own performance metric counters tailored to the business function. For example, for an online-sales application you could add a PMI counter for the number of items sold per day. Further you can extend the built-in performance monitoring and management capability with more extensive performance monitoring and management products from IBM Tivoli and third parties that also support the JMX standard.

The built in performance monitoring and tuning capabilities of WebSphere Application Server allows not only initial deployment tuning but also the flexibility to adapt on demand to changing business needs using real-time feedback from the production application.

A Leading Technology Partner

WebSphere Application Server, Version 6.1 delivers a robust and comprehensive infrastructure for SOA. When selecting a strategic foundation for your business, there is more to the decision than selecting a product – it is also about selecting a strategic business and technology partner. There are three main reasons why IBM is the premier strategic partner for SOA: standards leadership, product leadership, and proven experience.

- *Leadership in open standards:* IBM is committed to open standards to provide the greatest flexibility, interoperability, and investment protection. IBM continues to be the clear leader in supporting industry standards and has a tremendous history of leadership in standards development. IBM has led or co-authored the development of many of the key standards for J2EE, XML, and Web services. For example, WebSphere engineers have contributed to more than 80 percent of the J2EE specification, and these engineers continue to drive the evolution of J2EE standards through the Java Community Process. Similarly, for Web services and SOA, IBM continues to lead open standards development. IBM co-authored the WSDL and SOAP 1.1 specifications, which are the underpinnings of Web services today. Similarly, IBM authored the original UDDI specification and co-founded UDDI.org. In the area of Web services interoperability, security and transactions, IBM has been a leader as well, as the founder and chair of the WS-I organization, chair of the WS-I Basic Profile 1.0 working group, co-author for the standards for Web Services Security, Web Services Transactions, and the submission of Business Process Execution Language for Web Services to OASIS, to name just a few. IBM was the founder of Eclipse.org and provided the Eclipse platform to open source, which now provides the common tools interoperability framework for most J2EE vendors. Similarly, IBM was instrumental in the formation of OASIS, the industry consortium responsible for many of the Web services/SOA standards.
- *Product leadership:* WebSphere Application Server incorporates IBM's core capabilities and expertise in building system software. These capabilities include transactional and security leadership; an ongoing focus on interoperability; IBM's heritage in delivering robust and assured messaging infrastructure, expertise in distributed object and component technologies; strengths related to Web services and XML; industry-best support for manageability (including synergies with IBM Tivoli enterprise security and management capabilities); and significant experience in the area of application integration and connectivity. For J2EE, Web services and SOA, WebSphere Application Server continues to lead. WebSphere was the first J2EE 1.4 compatible server, first with WS-I Basic Profile support, first with a Web Services Gateway, first integrated UDDI directory, and first to leverage the Enterprise Service Bus to integrate applications and services across your organization and beyond with ease. WebSphere is leading with Web services support above and beyond initial specifications with additional programming model enhancements to provide a secure foundation for a SOA. Similarly, in the areas of rapid development and deployment, WebSphere product leadership is evident with frameworks that speed and simplify development and deployment such as Service Data Objects (SDO) and the rapid deployment facility. With the broadest platform support (including z/OS), the same core application server programming model across configurations¹, and connectivity and integration for enterprise systems, WebSphere Application Server, Version 6.1 leads the industry in delivering business flexibility,

scalability, high availability, and extending high-performance, proven application environment assets to your SOA.

- *Proven experience:* IBM has long heritage and extensive experience in transactions, security, and in developing extremely scalable and highly available applications and systems. With WebSphere Application Server, IBM is bringing the benefits of this experience to SOA applications, with a highly scalable transaction engine, leading security, world class clustering, and high availability management. Nobody has broader or deeper experience, or invests more in these technologies than IBM. IBM is investing over \$1 billion a year in SOA and Web services, and has tens of thousands of developers actively working on the WebSphere software platform and SOA-based offerings. In addition, IBM has over 10,000 IGS technical practitioners trained on WebSphere, providing a breadth and depth of expertise to assist customers that cannot be matched by any other vendor. With hundreds of SOA engagements under the belt, IBM has extensive real world experience with helping enterprises transform their businesses with SOA for on demand operations. This experience in helping customers implement high volume, highly available WebSphere applications feeds right back into the product development process. WebSphere Extended Deployment is a clear example of this, delivering to all customers many of the enhancements in terms of virtualization, high availability, workload management, application partitioning, and support for mixed workloads that were developed as part of IBM's direct experience in customer implementations. IBM's proven experience in helping customers implement a SOA and implement highly scalable and available enterprise systems means that you can mitigate your risk in adopting these technologies, leverage the best practices IBM has developed through actual engagements using current products, and improve your time to value.

For More Information

Contact your local IBM representative today for more information about how you can achieve business results well ahead of your competition.

Visit <http://www.ibm.com/websphere> for more information about WebSphere Application Server software and solutions.



© Copyright IBM Corporation 2006

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America. All Rights Reserved.

The e-business logo, e-business on demand, @server, CICS, DB2, Domino, IBM, the IBM logo, IMS, Lotus, OS/390, Parallel Sysplex, Redbooks, SecureWay, Tivoli, WebSphere, System z and z/OS are trademarks or registered trademarks of International Business Machines in the United States, other countries or both.

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

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

HP/UX is a registered trademark of HP in the United States and other countries.

Solaris is a registered trademark of Sun Microsystems, Inc. in the United States and other countries.

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

Linux is a registered trademark of Linus Torvalds.

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

All statements regarding IBM future direction or intent are subject to change or withdrawal without notice and represent goals and objectives only.

About Branham Group Inc.

Branham Group Inc. is a leading “Go-to-Market” consulting firm servicing the global information technology marketplace. Branham conducts work in the United States, Europe, Latin America, Asia and Canada. Branham Group assists information technology companies and related institutions in achieving market success through its planning, marketing and partnering services. Branham has a strong legacy of researching and analyzing vertical, horizontal and cross-industry opportunities. In Canada, it is well known for publishing the most comprehensive listing of the top Canadian ICT vendors (www.branhamgroup.com).

Branham Group, Branham300, and the Branham Group logo are the properties of Branham Group Inc.