WHITE PAPER

WebSphere Application Server Version 6.1: Advancing SOA for Greater Business Flexibility

April 2006

IDEAS Custom Consulting Services

PREPARED FOR

IBM

TABLE OF CONTENTS

Executive Summary

Market dynamics, combined with the continued introduction of new, enabling technologies, are driving the need for business flexibility. Most business leaders are seeing increased competition and new players with innovative business models entering their industries. Business success and even survival depends on access to real-time information and the ability to respond up and down the value chain to changing conditions. Business flexibility and profitable revenue growth are the watchwords of today's business executive.

Business flexibility refers to the ability to manage market volatility and unpredictability more effectively through the creation of flexible business models, processes, and infrastructures. By enabling businesses to cope with constant change, these flexible structures afford competitive advantage and drive business value. Increased business flexibility enables organizations to become "on demand" businesses. An on demand business is an enterprise whose business processes – integrated end-to-end across the company and with key partners, suppliers, and customers – can respond with speed to any customer demand, market opportunity, or external threat. Business leaders recognize that they need to sense, analyze, and respond more effectively to continuously changing market conditions and risks. Service-oriented architecture (SOA) represents a critical enabler of responsiveness and business flexibility; lagging behind in this area denotes a serious competitive weakness.

THE IDEAS BOTTOM LINE

- Market dynamics and the continued influx of new, enabling technologies have accelerated competition and intensified the need for greater flexibility in terms of IT's support of business goals and objectives.
- SOAs are the most likely choice for supporting the creation of the robust, flexible, and scalable infrastructures – required by enterprises of all sizes to create and maintain competitive advantage. A comprehensive software "stack" of well-integrated infrastructure elements is generally required to build, deploy, and manage SOA-based solutions.
- With IBM WebSphere Application Server Version 6.1, IBM has made significant enhancements that improve the product's ability to serve as the foundation for on demand solutions with high business flexibility. The product better facilitates efficient development and deployment; provides a higher performance SOA runtime with improved security, scalability, and availability; offers an enhanced communications infrastructure; and features streamlined application management and maintenance capabilities. All of these enhancements contribute to making Version 6.1 an even better choice when SOA enablement and business flexibility are critical solution criteria.

As part of its continuing effort to enable businesses to meet the increasing demand for flexible solutions, IBM has recently introduced WebSphere Application Server Version 6.1. This new version is built to better deliver the secure, scalable, and resilient application infrastructure needed for SOAs. More specifically, Version 6.1 brings new functionality that:

- significantly improves the efficiency of development and deployment;
- enhances the level of security and eases the task of implementing security mechanisms;
- provides greater performance due to new J2SE 5.0 support and improvements to the Web services engine and messaging infrastructure;
- improves service reusability and accessibility through new communication services; and
- reduces the resources and costs associated with management and maintenance tasks through enhanced application management functionality.

With this latest announcement, IBM further bolsters its already strong position as a vendor able to deliver a platform and infrastructure that helps its customers fully exploit the business flexibility of SOAs.

The Case for Business Flexibility

Several factors have had a tremendous influence on IT's role within the enterprise over the past ten or so years. Driven by enabling technologies (the Web, XML, Java, and others) that empower organizations to more aggressively leverage their IT assets to gain competitive advantage, IT has found itself in a position it has not been in before – one in which its output is tied directly to the bottom line more strongly than ever.

For IT managers, this has required not only gaining a better understanding of the business environment in which they work, but also working more closely with those directly responsible for revenue and profits. At the same time, business managers have had to become more familiar with how new technologies help them do business more efficiently and with greater reach into the communities with which organizations must interact.

Naturally, these changes have created a more challenging and complex IT environment in which technologies must be mastered and integrated with existing systems. The ultimate goal is to build comprehensive systems for an "on demand" age. From the business perspective, a growing recognition of the potential benefits of leveraging such systems has resulted in a highly competitive landscape in which success is measured in terms of efficiency, flexibility, and customer satisfaction.

Ironically, increased complexity and greater technology challenges have been met with an unabating push to deliver greater value for less money and in less time. Driving this efficiency mindset are:

- the increasingly intense competitive environments in which enterprises find themselves;

- the end of the Internet "boom," which has forced organizations to focus more attention on more traditional business metrics and modes of accountability; and
- a realization that IT budgets can quickly get out of control given IT's burgeoning role.

Therefore, today's enterprises face an increasingly intense and rapidly changing competitive landscape, instigated in great part by the availability of technologies that permit tight integration with partners, suppliers, and customers at the business process level while at the same time increasing the complexity of building, deploying, and managing solutions. The IT infrastructure on which an organization relies must be flexible enough to allow it to respond to continuous and major change quickly, cost-effectively, and with minimal disruption to the business operations and activities it supports. In order to create the required business flexibility, an enterprise must accomplish four major tasks:

1. The enterprise must align its business models and processes with its strategic goals and objectives. While not strictly an IT issue, this alignment is a necessary precursor to leveraging IT to meet business objectives.
2. Once the above alignment is achieved, the enterprise must then standardize and automate its business processes by integrating them with the underlying IT infrastructure. The enterprise must work with vendors that provide the standards-based tools and technologies that support business process automation and the types and levels of integration required.
3. To create efficiencies and minimize disruption to its business, the enterprise must transform its existing IT assets to achieve modularity and reusability.
4. The enterprise must build the ability to scale quickly and cost-effectively into the resulting IT infrastructure in order to meet the growing demands of the business.

The IT infrastructure is the technological foundation for a true "on demand" business, and as such, it must be able to provide the required application functionality where and when it is needed. It must do so through effective and flexible integration of IT assets, and an efficient development model that eases the burden of adopting and using new technologies.

The Role of SOA in Achieving Business Flexibility

The introduction of Web services in the 1990s was a major step in the long and sometimes slow history of creating interoperability across a heterogeneous world of platforms, software technologies, and proprietary assets. These assets were seldom designed for such interoperability, but enterprises had too much invested in them to reinvent the proverbial wheel.

With Web services (or, more accurately, Web services standards), the chore of integrating disparate assets and sharing data and information could be abstracted away from the assets themselves. With a finite but controllable investment in creating Web services interfaces for existing functionality, IT could achieve greater degrees of interoperability than ever before. Perhaps more importantly, because the Web services model was founded on the notion of modularity, discrete functions could be assembled and reassembled virtually in real time when needed to meet the customized needs of a given service consumer.

Thus, Web services laid the foundation for an agile and cost effective application infrastructure that could help realize the vision of network computing: linking an enterprise with the various communities with which it conducts business in a flexible, responsive, reliable, and secure way. Whether it was automating the process of negotiating a deal with a supplier, or enabling a customer to order products online, Web services held the key to leveraging business systems to support business processes and transactions – the essence of "on demand" computing.

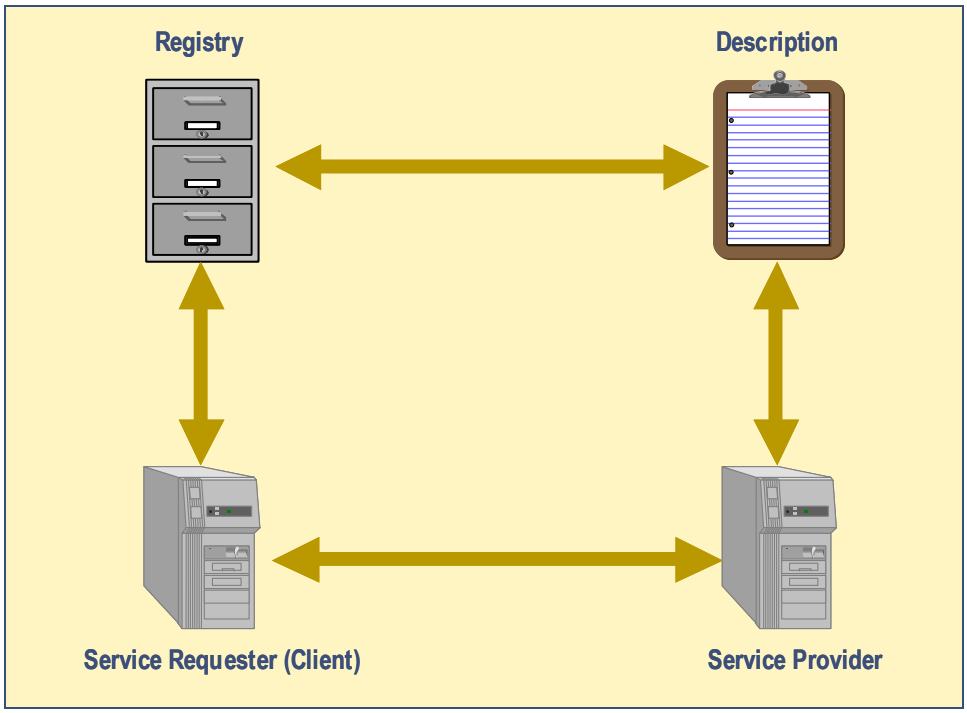
However, while the Web services model certainly establishes the standards and development paradigms for building flexible on demand systems, an architectural context for their deployment, operation, and management is also needed. The response takes the form of service-oriented architectures (SOAs). Figure 1 graphically depicts an example of an SOA. While SOA definitions vary, there are a number of common and foundational elements that give SOAs their unique characteristics:

1. A methodology for creating service descriptions in the form of documents that specify where services are located, how to access them, and what their exposed operations and/or methods are. This methodology includes a language specification (Web Services Definition Language) used for descriptions.
2. A registry to which a service provider can publish descriptions and that implements access control mechanisms used by service descriptors to find the services they need.
3. Binding and messaging mechanisms that permit a service requestor to tie the discovered service to a service requestor's application or service.
4. Access control mechanisms to authenticate and authorize service requestors.

Coupling is loose and dynamic, creating a high level of flexibility and agility.

FIGURE 1

How an SOA Works



Implementation of an SOA requires a variety of functional elements that make up a software "stack." This stack must support a variety of Web services standards (as well as others), and implement a messaging infrastructure that supports the XML-centric requirements of SOA enablement as just described. Security is, of course, a primary concern, and therefore the stack must contain robust security functionality that achieves the required levels of access control and intrusion protection. An SOA's underlying infrastructure must be implemented in a scalable fashion to grow as the business grows. Finally, tools and deployment functionality must accommodate rapid and cost-effective change, supporting developers and administrators in ways that minimize the resources required to build, deploy, and manage applications. Only in this way can the enterprise respond as it must to changing market and competitive circumstances.

IBM WebSphere Application Server Family: Delivering a Flexible Application Infrastructure

The WebSphere Application Server family, in its various configurations, forms the foundation for the WebSphere software platform. WebSphere Application Server is one of the most widely used Java-based application platforms (with full J2EE 1.4 compatibility), and provides a highly functional application deployment environment that can be used to integrate enterprise data and transactions. The broad variety of product configurations included in the family enables WebSphere Application Server to support a wide range of deployment options – from single-server implementations to clusters that support high-volume transactions and high availability. The products leverage rapid development and deployment features that enable organizations to bring new applications and enhancements online quickly and efficiently, while optimizing their existing skill-sets and resources.

WebSphere Application Server is one of a variety of products that support IBM's ability to deliver an SOA reference architecture that supports enterprises' efforts to implement fully functional SOA implementations. The reference architecture (depicted by IBM in Figure 2 below) contains services to support connectivity to relevant communities (partners, customers, etc.), business process orchestration, data and information management, collaboration, secure access, business decision-making, services management, application development, and of course the business applications themselves. The architecture has as its foundation a set of infrastructure services designed to optimize throughput, performance, and availability.

WebSphere Application Server Value Proposition

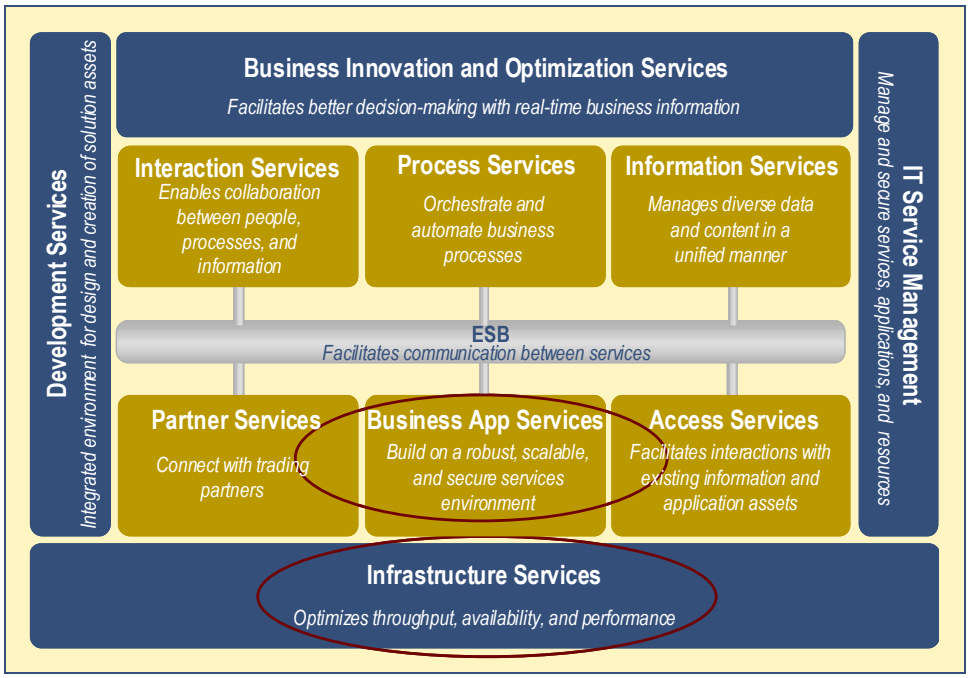
The features and functionality of the WebSphere Application Server family are designed to enable greater business flexibility. The WebSphere Application Server programming model includes a set of consistent services for developing transaction-oriented applications and provides the foundation for integrating other applications, data, and business processes. IBM delivers a flexible application infrastructure that provides:

- A secure, scalable, and highly available SOA runtime that optimizes uptime for mission-critical applications; optimizes resource utilization; keeps key IT assets safe; and scales easily and quickly as the business grows.

- Leading support for standards-based messaging and Web services (including Java and XML standards), which facilitates the use of composite applications and provides the message transport mechanisms needed to support SOA-based solutions.
- Extensive communication services that enable businesses to make application services more reusable and accessible to new users in new ways.
- Rapid development and deployment features that simplify and accelerate the development process, thus improving time to value while leveraging existing technology skills. Businesses that leverage these features can achieve the flexibility needed to grow and differentiate as business requirements evolve.
- Simple and effective application management, which can free up company resources and lower management and maintenance costs. These extra resources can be repurposed toward innovation for competitive advantage.

FIGURE 2

IBM's SOA Reference Architecture



The WebSphere Application Server Family

The WebSphere Application Server family (as depicted in Figure 3 below) offers a range of configuration options and supports multiple business models and deployment platforms. This broad portfolio enables IBM to satisfy a variety of customer needs and usage scenarios. Configuration options include:

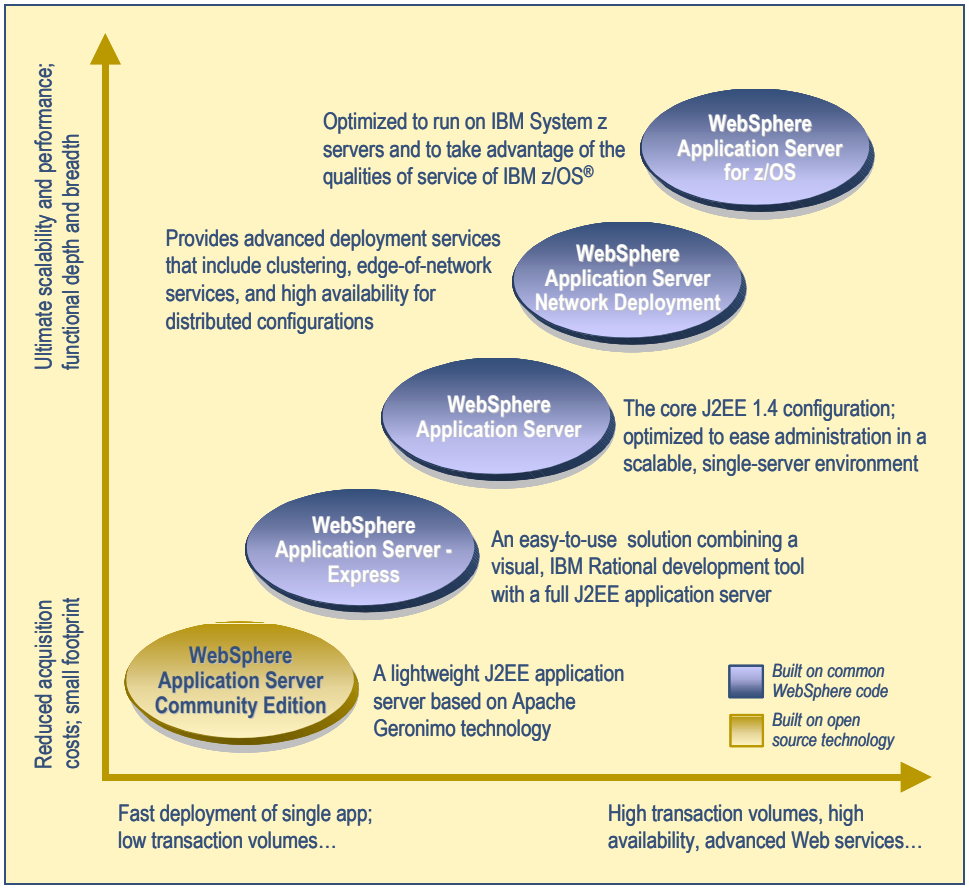
WebSphere Application Server Community Edition. WebSphere Application Server Community Edition V1 is an entry-level offering built on open source technology from the Apache Software Foundation. The product's intent is to bring the innovation and cost-saving benefits of open source community development and distribution to the WebSphere family, giving Java developers a highly accessible and flexible platform for building and deploying Java applications. The product is

easily downloaded directly from IBM's Website and is free to use in development and production. Customers pay only for technical support from IBM.

The code base for WebSphere Application Server Community Edition is obviously different from the rest of the offerings in the product line. Since migration to higher-level product configurations is a path customers of this edition might take, IBM has committed itself to make such a migration as seamless as possible.

FIGURE 3

The WebSphere Application Server Family



WebSphere Application Server – Express. WebSphere Application Server – Express[1] is an easy and affordable entry point that combines a visual development tool from IBM Rational with a fully J2EE-compatible application server. Targeted primarily at small and medium-sized businesses, ISVs, and corporate developers building what IBM refers to as "direct to middleware" applications, WebSphere Application Server – Express provides an out-of-the-box solution that facilitates the construction and deployment of static and dynamic Web pages and Web applications. A set of integrated applications, wizards, and samples help users get up and running quickly, and the tight integration between the development environment and the application server enables simpler and faster creation of dynamic Websites.

[1] WebSphere Application Server – Express Version 6.1 will be delivered in the third quarter of 2006, in line with the new Rational tools that are to be included with the product.

WebSphere Application Server. WebSphere Application Server, the core offering of the family, is an application platform optimized to ease administration in a scalable, single-server deployment environment. This core J2EE 1.4 configuration delivers a secure, high-performance transaction engine that is well suited for standalone, departmental applications and Web services.

WebSphere Application Server Network Deployment. WebSphere Application Server Network Deployment is IBM's primary application server offering for businesses and partners who need a full J2EE application platform with higher qualities of service built in. This configuration adds advanced deployment services to the core WebSphere Application Server functionality, including clustering, edge-of-network services, and high availability for distributed configurations. Because of its focus on network-based configurations, WebSphere Application Server Network Deployment contains support for intelligent workload distribution across a cluster; failure bypass; remote administration of firewalls from a browser-based console; and simplified administration. In addition, Edge Components (previously known as the WebSphere Edge Server) are included with the product, and provide advanced support for load balancing, caching, and centralized security. WebSphere Application Server Network Deployment also provides advanced support for Web services with the Web Services Gateway, which enables Web services invocation by users from outside the firewall with the benefit of robust security protection.

Clearly, IBM's strategy here is to position WebSphere Application Server Network Deployment as the choice for enterprises requiring near-continuous availability as well as advanced management and automated performance optimization for mission-critical applications. This edition is designed to meet the needs of enterprises with highly scalable and available network-based implementations that often reach beyond the walls of the enterprise as their main mode of operation. With greater capability planned for IBM's WebSphere Application Server Network Deployment offering, IBM is positioning Network Deployment as the universal foundation for enterprises. Network Deployment will remain the high availability foundation and will feature continued improvements in this area.

WebSphere Application Server for z/OS. The z/OS version of the WebSphere Application Server delivers a J2EE application server specifically optimized to utilize the unique qualities of service provided by IBM z9 hardware and the z/OS operating system. The programming and deployment models of WebSphere Application Server for z/OS are functionally equivalent to those of WebSphere Application Server Network Deployment, providing for greater flexibility in development as well as improved deployment and management functions. WebSphere Application Server for z/OS also provides optimized transactional integration with CICS, IMS, DB2, and VSAM. It is designed to bring a variety of enterprise class capabilities – including efficient access to enterprise data, prioritized workload management, complex transactional integrity, high levels of scalability, and mainframe-class security – to Java and Web services-based applications.

WebSphere Extended Deployment. While not an application server per se, WebSphere Extended Deployment is an important extension of the product line. It has three primary roles:

1. To help users implement a "scaling on demand" approach through dynamic allocation of server hardware resources to applications. This approach offers levels of scalability suitable for high-end transactional environments to a wide array of IBM and non-IBM platforms.
2. To provide simpler and better management of complex system operations with meaningful, real-time visualization tools and a gradual, controlled implementation of autonomic capabilities.
3. To support new application types and workload patterns. Transactional and long-running workloads can now be consolidated into a single application infrastructure called a Business Grid. The result can be enhanced quality of service levels for the J2EE platform on all the systems an enterprise employs.

WebSphere Application Server, Version 6.1 Enhances IBM's Ability to Deliver Business Flexibility

WebSphere Application Server was created by IBM to help its customers build multitier solutions using industry-standard technologies. Since its introduction in the late 1990s, it has grown to become one of the most widely used application server products available, and has been enhanced in a variety of different ways to meet the growing needs of "on demand" enterprises.

With the introduction of WebSphere Application Server Version 6.1, IBM takes major steps toward enhancing the business flexibility required today by enterprises of all sizes. IBM has added functionality and improvements designed to:

- make development and deployment of new functionality and applications even faster and easier;
- improve security, scalability, and availability for a more robust and reliable SOA runtime;
- improve the accessibility and reach of application services through extensive communication services; and
- reduce the costs and resources associated with maintenance and management through new application management capabilities.

The enhancements introduced in WebSphere Application Server Version 6.1 are built on the foundation laid in earlier versions of the product. IBM's road map to create a platform that enables enterprises to achieve higher levels of business flexibility and competitive advantage is being further fulfilled with this latest version of WebSphere Application Server.

Simple, Rapid Development and Deployment

The imperative of business flexibility is in great part served by the ability to quickly respond (from an IT perspective) to changing market dynamics and competitive forces. As noted earlier, a growing need to "do more with less" is driving IT

organizations to adopt products and tools that automate as much as possible the development of new functionality. These tools must also leverage to the greatest degree possible the existing skills of developers.

One way IBM is enhancing the development environment for WebSphere Application Server Version 6.1 is by adding support for the Java 2 Standard Edition (J2SE) Version 5.0 specification. Codenamed "Tiger" when it was first conceived, J2SE 5.0 contains a variety of features intended to ease the development task. For instance, the specification includes a new metadata feature that allows the association of additional data alongside interfaces, method fields, and Java classes. This additional data, also known as "annotations," can be read by tools (including the javac compiler) and, if the configuration permits it, be stored in the class file and discovered at runtime. In this way, tools used to develop or deploy applications to run in the J2SE virtual machine can leverage a common infrastructure and therefore potentially reduce the resources needed to program and deploy those applications.

Another enhancement introduced as part of J2SE 5.0, and supported by WebSphere Application Server Version 6.1, is generic types. High on the list of desired new functionality by Java developers for some time, generic types permit the designer of an API to create common functionality that can be used with multiple data types and type-checked at runtime. By eliminating some of the need for casts and improving runtime type-checking, generic types help to increase developer efficiency.

J2SE 5.0 also includes support for the autoboxing and auto-unboxing of primitive types. This support eases the chore of converting between primitive types and their object-based counterparts by producing code that is more concise and easier to understand. There are a variety of other enhancements – such as support for Enumerated Types, which offers developers a more elegant means to declare a sequence of named constants – and other features that contribute to the efficient use of development resources as well.

IBM has also introduced innovations that complement the features of J2SE 5.0 described above. First, enhancements to its virtual machine capabilities make it easier to deploy on a wide range of devices – from handheld devices to larger servers. The architecture is designed to be both modular and pluggable, making it flexible and reusable in a variety of runtime scenarios. Better profiling and debugging capabilities – such as full speed debug and hot-code replacement – can help accelerate the diagnosis of problems. Dump and trace engines are provided that can help to improve the serviceability, as well as the reliability, of applications. IBM also includes a fine-grained locking mechanism of virtual machine data structures, which helps to increase performance by reducing the amount of synchronization needed.

IBM has made some Just-In-Time (JIT) Compiler enhancements as well. The compiler leverages the mechanisms of J2SE 5.0 (more specifically, Java Development Kit 5.0) to optimize compiler performance. Previously, the compiler only offered two optimization levels. With this latest version, there is a dynamic and more granular selection heuristic that can limit the time spent on compiling less important parts of an application. In addition, Java methods can be compiled

asynchronously (on a background thread), which can help improve the startup time of applications with a high level of multithreading on SMP servers. Finally, IBM has made some enhancements to its garbage collection capabilities that are intended to lower latencies and reduce the number of long pauses in the JVM runtime environment. The capabilities to both balance the garbage collection activity across threads, and to place objects into "new" and "old" spaces based on usage, can make the overall process more efficient. Garbage collection for large SMP machines can be "sub-pooled" to improve application throughput.

Not surprisingly, the primary motivation for IBM to shift the WebSphere Application Server runtime to the J2SE 5.0 implementation is to help IBM's customers more efficiently develop and modify their own applications. These customers can use these features with few constraints. However, because the J2EE 1.4 specification does not yet support the new language features discussed above, IBM advises customers not to use generic types with public EJB interfaces that are exposed on the home, on stubs, and so on. In addition, deployment of applications that use the J2SE V5.0 language features are currently restricted to WebSphere Application Server Version 6.1 – earlier releases do not support, or provide, the V5.0 virtual machine.

One important and interesting change in Version 6.1 is in the way that IBM supports JSR 168. Developed using the Java Community Process (JCP), JSR 168 provides a standard API for building portlets. With Version 6.1, IBM has moved the JSR 168 portlet container from the IBM Portal Server down into the WebSphere Application Server in order to provide a common programming model for presentation logic across all of its application server-based offerings. From the programming standpoint, customers no longer have to choose between servlet and portlet programming models when creating a presentation layer. Instead, they can code to the JSR 168 portlet open standard and then run the resulting code on all WebSphere Application Server-based offerings. In this way, the need to potentially transform or rewrite presentation logic when moving from one WebSphere Application Server offering to another is eliminated.

To make WebSphere Application Server deployments easier, more reliable, and more repeatable, IBM provides IBM Installation Factory for WebSphere Application Server. Delivered as part of WebSphere Application Server, it is designed to streamline the process of getting an application server installation up and running by reducing the task to one step. The Installation Factory leverages an Eclipse-based GUI and command-line interface, and allows developers to put together a WebSphere Application Server release (including available patches) for the purposes of doing a custom install to the customer's particular environment. With Version 6.1, IBM has added the ability to generate cross-platform installation packages (specifically, any UNIX/Linux to any other UNIX/Linux or any UNIX/Linux to Windows). The release also permits the importation of predefined sets of configuration information from a "configuration archive" into a newly created profile, which not only saves time and effort but also promotes reuse and reliability. Installation Factory also supports the deployment of EAR files and the execution of scripts.

As a key part of WebSphere Application Server Version 6.1, IBM is shipping an enhanced version of the Application Server Toolkit. The toolkit provides new

wizards and tools for creating Web applications, Web services, portlets, and EJBs. It supports annotation-based programming (a J2SE 5.0 feature described earlier), new administration tools, and tools to edit WebSphere-specific bindings and extensions. It also provides a new Unit Test Environment that includes a Unit Test Client Web application to facilitate testing of EJBs and Web services. Other features supported by the Application Server Toolkit, such as Session Initiation Protocol (SIP) servlet support and some new administration capabilities, are described later in this white paper.

Reuse can be a significant enabler to development efficiency and the reliability of the resulting applications. One important enhancement IBM is introducing with Version 6.1 is the inclusion of a Java Server Faces (JSF) widget library. The library is basically a set of prebuilt functions that can be used to build Web user interfaces. Examples of widgets include Graph, Input, Menu, Tree, DataGrid, and others.

From the tools perspective, IBM intends to provide tight integration between its Rational tools and WebSphere Application Server Version 6.1. IBM intends to ship a new version of Rational Application Developer later this year, and to likewise offer a new Rational toolset as part of the Version 6.1 release of WebSphere Application Server – Express.

IBM has clearly made developer efficiency an important part of its WebSphere Application Server Version 6.1 release. It has combined improvements and enhancements in Java specifications with its own product creation capabilities to bring a more productive development environment to WebSphere Application Server users.

Making the SOA Runtime More Secure, Scalable, and Available

As we noted earlier, the ability to react quickly to changing market conditions is an imperative in today's business world, and services-based architectures, as instantiated in the concept of the SOA, are an effective means of harnessing business agility. Needless to say however, rapid development will not provide benefits if reliability – and security – cannot be assured.

With WebSphere Application Server Version 6.1, IBM has introduced a variety of new enhancements aimed at improving performance in these two major areas:

Security. While an agile infrastructure is required to permit interoperability within and across enterprises, it exposes organizations to a potentially long list of access points and users. This increased exposure of IT assets raises the importance of protecting critical business information from unauthorized access and misuse. IBM's approach to security is to deliver an open standards-based security infrastructure that will protect the environment from a variety of potential threats, yet still enable the enterprise to fully leverage SOA advantages. A key part of the security infrastructure is based on the WS-Security specification, which has been recently updated, and the Web services Interoperability Basic Security Profile. This profile, developed with the Web services Interoperability Organization (WS-I), is a set of open Web services specifications that address security in terms of the transport layer, SOAP messaging, token profiles (username, X.509 certificate),

XMLSignature, XMLEncryption, and other important security mechanisms. Support for this profile helps enable stronger and tighter interoperability among runtime platforms from different vendors (for example, it securely allows a WebSphere client to call a .NET server). In addition to these first steps in ensuring full security interoperability among application servers from different vendors, new WS-Security mechanisms enable increased portability of applications across application servers as well. At a higher level, this support for enhanced security mechanisms has allowed IBM to achieve Common Criteria Assurance Level 4 certification. Also known as ISO/IEC 15404, the Common Criteria is basically a framework in which user requirements, vendors' product claims, and evaluations can coexist within a process context to determine whether specific products can be effective in meeting user needs. While not the highest assessment possible, meeting level 4 requirements makes IBM highly competitive.[2]

IBM has added other security improvements in Version 6.1 as well. One example is support for networks that handle cross-platform authentication through the Simple and Protected GSSAPI Negotiation Mechanism (SPNEGO). This authentication extension to "HTTP Negotiate" enables tokens that conform to the SPNEGO specification to be passed from a browser-based client on one platform to a server based on a different platform using HTTP headers. This improvement will likely have its greatest applicability to the scenario in which IE browsers pass tokens to UNIX- or Linux-based Web servers. With Version 6.1, default security configurations are set "out of the box." Hence, they are available and online as soon as the product is installed – prior to customized configuration to the specific environment. To support effective identity management, a default user registry is shipped with Version 6.1 as well.

Scalability and Availability. Business agility requires flexibility that not only supports fast and cost-effective change to the infrastructure and applications, but also the ability to scale as the business grows. Scaling the environment may involve a number of initiatives, one of which is improving the performance of existing assets. For this, IBM has included a number of important new capabilities in WebSphere Application Server Version 6.1. It is important to note at the start that WebSphere Application Server products are based on IBM's J2EE 1.4 compatible runtime environment; continued improvements to the J2EE specification over time have meant improved performance for those products that support it (IBM included). Performance improvements have also been achieved via IBM's implementation of J2SE V5.0 – specifically, the use of JDK 5.0 and the enhancements described earlier in this paper hold the potential for significant runtime performance improvement. One interesting feature of the JVM is the use of shared classes, a useful feature in server environments in which multiple JVMs exist on the same box. Shared classes eliminate the need to load all classes for each JVM instance, and the use of a shared memory cache can result in faster and reduced memory usage.

Other performance benefits have been achieved through improvements in the messaging infrastructure built into the WebSphere platform. These will be discussed in more detail in the next section of this paper. In addition, some of the

[2] The certification process requires the general release of stable code, after which required paperwork may be submitted. Certification is therefore expected soon after general release.

performance improvements in Version 6.1 are attributable to enhancements to the product's Web services engine. 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 therefore scalability. A faster parsing technology also adds to performance.

An important contributing factor to scalability and availability is platform support. With its broad platform support, IBM continues to do well in this regard, enabling customers to add capacity using systems that are right for their specific needs.

Through new security features, a new Web services engine, support for JDK 5.0 and other standards, IBM is now able to bring to its customers a more robust runtime to support their efforts to build and deploy solutions based on SOAs.

Extensive Communication Services

Developing and deploying new applications and services are obviously critical steps in the process of creating business flexibility through IT investments. Optimizing return on that investment can often require making those application services more reusable, as well as more accessible (to a wider variety of users and in a greater number of usage scenarios). Doing so requires a sophisticated set of communication mechanisms that can extend the reach of application services beyond the boundaries of their enterprise and their internal IT infrastructure, as well as the types, formats, and usages of information delivered to and from those services.

Perhaps the most important change IBM has made to the communications infrastructure of WebSphere Application Server Version 6.1 is support for Session Initiation Protocol (SIP) servlets. Developed by IETF MMusic Work Group, SIP incorporates a protocol standard for the initiation, modification, and termination of interactive user sessions involving multimedia content (video, voice, instant messaging, online games, and virtual reality). The relevant specification (JSR 116) enables the implementation of SIP using the Java servlet model.

Today, networks that utilize SIP are employed by ISPs and enterprises to implement chat and instant messaging solutions. SIP can also be used to support videoconferencing, IPTV, and other technologies, and it is likely these will see the light of day in the not too distant future. However, perhaps the most visible example of an SIP network is one that supports voice-over IP (VoIP). With WebSphere Application Server Version 6.1, IBM provides SIP servlets as well as a unique piece of functionality called the Converged Servlet Container. This container permits a variety of servlet types (SIP, HTTP, and portlets, for example) with disparate protocols to share the same application session and tie their states together. Hence, SIP functions can be seamlessly built into applications, and the state of a particular activity that uses the SIP protocol can be visible to that application. This capability is best illustrated with an example: A CRM application used by a client services employee can contain an icon (perhaps presented through a portlet) that the employee can click to initiate a call with a customer. The SIP servlet will initiate and control the call within the CRM application and portlet, with state information available to all three elements. SIP tooling is provided in the Application Server Toolkit, making it possible for developers to easily build SIP applications.

Many of the additional communications services IBM has built into Version 6.1 are reflected in enhancements to the Web services engine that is part of the product. One of these is support for Web Services Notification (WS-N). This specification is an event-driven (notification-based) pattern for interaction based on interobject communications. It represents a Web services-based instance, or adaptation, of publish/subscribe models used in scenarios involving message-oriented middleware. By standardizing the way Web services interact in this manner, the interoperability between Java and non-Java environments can be improved. Since we expect event-driven architectures built using Web services to grow in importance over time, WS-N support will become more of an imperative. IBM has it today in WebSphere Application Server Version 6.1.

Transactions often form the basis of interaction among enterprises, and are in fact the major focus of IT systems used to support business activities conducted with partners, suppliers, and customers. IBM has added support for Web Services Business Activity (WS-BA) to WebSphere Application Server Version 6.1. The WS-BA specification defines a business activity coordination type used with the extensible coordination framework described in the WS-Coordination specification. Functionally, WS-BA provides a framework through which application code can be automatically invoked to compensate for a transaction that has already been committed (in situations in which the wider scope of work is deemed to have failed). This capability enables traditional two-phase transactions to be completed in a timely manner (thus releasing database locks) while still providing a way for the application to logically “undo” that work if necessary after subsequent processing. The ability to commit transactions and release locks in this manner, while still retaining the option of compensating for that work if necessary, is particularly important as Web service requests begin to span trust boundaries across departments and enterprises, and will be a key capability for many SOA applications.

Improvements to the messaging capabilities are also an important part of Version 6.1. One such enhancement is focused on SOAP messages passed to and from Web services using Java. The concept of Web services relies on the ability to send and receive messages in a standardized format understandable to all systems involved. The most prevalent message format in this case is SOAP, but a specification called SOAP with Attachments API for Java (SAAJ) can help 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 (JAXM), SAAJ can result in performance and productivity enhancements over generating messages manually. Recent changes to SAAJ have improved the mechanisms it provides to users, potentially making performance benefits even greater. IBM has also improved JMS performance with changes to SOAP/JMS.

IBM has been committed for some time to continuously improving the integration between WebSphere Application Server and other IBM messaging products. With Version 6.1, IBM has improved the integration between IBM WebSphere MQ for z/OS and the default messaging provider in WebSphere Application Server. For instance, users can add a WebSphere MQ for z/OS queue manager or queue sharing group as a member of a service integration (SI) bus, potentially improving availability and simplifying administration tasks. Additionally, users can employ the

shared queue capabilities of WebSphere MQ for z/OS and the mediation capabilities of IBM WebSphere Message Broker for z/OS from an SI bus. Version 6.1 also offers a choice of message store types used in the messaging engine – data objects can be stored in a database or in a file system. The file store option, which is new, can help improve performance and make configuration and management easier.

More Effective Application Management

Managing and maintaining applications and IT infrastructure has always been a time-consuming and resource-intensive chore. The technological changes and increased complexity that Web services introduce have driven the need for new tools and management products that can help IT organizations manage more efficiently, and as a result, free up resources to create new, innovative applications to better support business processes and activities.

Version 6.1 of WebSphere Application Server includes a number of new capabilities aimed at easing the management task. One example is a new Console Command Assistant that automates the task of creating scripting commands directly from actions taken by users within the administration console (calling applications, changing parameters, etc.). This is a powerful alternative to one introduced in WebSphere Application Server Version 6.0 that provided a drag and drop capability for modeling administration tasks onto a palette and then generating scripting commands. IBM also furnishes this capability in the Automation Toolkit shipped with Version 6.1 as a plug-in that brings automated administration scripting to the Eclipse environment.

The IBM HTTP Server for WebSphere Application Server is IBM's Apache-based Web server technology. Version 6.1 includes the ability to fully configure – and manage – the IBM HTTP Server through the WebSphere Application Server administrative console. In addition, security management has also been simplified through a variety of mechanisms:

1. Key management tools, and the management of Web server and plug-in certificates, have been built into the administrative console.
2. The TrustManager can be used to automatically trust hosts and signers.
3. IBM has made it easier to refresh an expiring certificate, and to understand and use the functionality that configures SSL attributes.
4. Administrative security is now enabled "out of the box," and configuration of security features has been made simpler.

IBM is also including a new standalone thin administration client with WebSphere Application Server Version 6.1, enabling customers to install an administration client and associated scripts on remote machines that have limited disk capacity. The client can be packaged either as a JAR file or as an Open Services Gateway Initiative (OSGi)[3] based bundle.

[3] An OSGi-based architecture is based on an open standard that provides for application server functional modularity. Since customer applications load only those application server components required, memory footprint requirements can be significantly reduced.

Clustering is an important technology to achieving the scalability and availability requirements discussed earlier in this white paper. In Version 6.1, IBM provides an enhanced cluster creation wizard that facilitates the process of designing clusters. For example, the wizard allows for a base image to be recreated on systems throughout a cluster, provides "cheat sheets" to aid in understanding, and reveals which server cluster members are available and not available. As a systems management enhancement, Version 6.1 allows for incremental cluster cell upgrades, removes some of the restrictions imposed with Version 6.0, and allows Version 5 and Version 6 server creation and coexistence within the same cell.

The IBM Support Assistant is now bundled with WebSphere Application Server, creating the equivalent of an "electronic service desk" right in the product. Previously, this functionality was only available as a Web download. Access to the Support Assistant is via the Web or through a standalone GUI. Important functionality includes:

- a Search Component that can search both IBM and non-IBM locations to find the most pertinent support information;
- Support Links that offer quick access to product and support pages, news groups, etc. that are relevant to a particular user issue; and
- a Service Component that facilitates the submission of service requests directly to IBM by automatically collecting and adding system data that IBM Support will need to perform the required analysis.

This assistance in managing problem reports and collecting data is designed to accelerate software problem resolution and therefore minimize business impact.

IBM has clearly made system and application management a high priority in the WebSphere Application Server Version 6.1 release. The company also likes to accentuate the role of an "ecosystem" of application servers, the applications they host, and the users who build and support the implementations. In addition, IBM complements its support offerings and ecosystem benefits with extensive documentation and training initiatives.

Conclusions

In delivering WebSphere Application Server Version 6.1, IBM has correctly addressed four major areas of strategic focus:

1. Simplifying development and deployment tasks
2. Enhancing its already robust SOA runtime with better security, scalability, and availability features
3. Augmenting and improving communications services to increase access and reusability by a wider range of communities
4. Creating a more efficient infrastructure for management and maintenance

All of these enhancements allow IBM to continue to position WebSphere Application Server as a leading choice to support SOA enablement and business flexibility.

One of the key values customers look for is a rich ecosystem with plenty of third-party support for the platform. IBM works to ensure that its channel is well equipped to deliver obvious value to customers. IBM provides the standards ISVs need to meet their development delivery schedules. ISVs and channel partners need a complete ecosystem in order to deliver value in a cost-effective and timely manner. This ecosystem includes all of the tools that partners require to sell their products, including demonstration and sample code, enablement sales tools, presales technical support, and a wide range of pricing options. IBM continues to focus on improving its partners' ability to compete and win.

Customers and partners alike see superior ease-of-use and a sound out-of-the-box experience. IBM's strategy is to convey a consistent look and feel throughout all of the components of the application server. IBM also continues to improve its WebSphere tools to speed up the time-to-value for customers. Improved time-to-value can mean a lower cost of ownership, especially in the area of administration, and can reduce the need for high-cost deployment skills. IBM's goal is to continue to reduce time-to-value and increase business flexibility for enterprises of all sizes.

Further, IBM is enabling an easy migration path across the different configurations of the application server. While there still may be some work to be done in terms of making this happen for users of WebSphere Application Server Community Edition, IBM has convinced us that it understands the importance of extending "seamless compatibility" in this case, and is committed to making it happen in a reasonable timeframe. In fact, Ideas International believes that IBM has created a strong open source application server presence with Community Edition. Achieving compatibility across the entire WebSphere Application Server product line is, from a competitive standpoint, an imperative.

We view IBM as a "platform" vendor – i.e., a vendor with a comprehensive software "stack" to support all of the functionality required for a successful SOA implementation. The WebSphere Application Server strategy is part of IBM's delivery of a complete software stack that includes a variety of tightly integrated middleware solutions and tools solutions. WebSphere Application Server is the foundation for the SOA-centric, on demand operating environment IBM offers that can provide quality-of-service levels tailored to specific customer requirements. Perhaps more importantly, the WebSphere Application Server family, combined with the rest of the software stack, as well as the company's grid and virtualization capabilities, can easily scale with customer needs – a major manifestation of business flexibility. All of this functionality is available from a single vendor, easing the purchase, delivery, deployment, and support of business solutions.

Ideas International therefore sees IBM as capable of continuing a leadership role among infrastructure vendors, particularly where SOA enablement and business flexibility are key elements of a customer solution.

Americas
Ideas International, Inc.
800 Westchester Avenue
Suite S620
Rye Brook, NY 10573-1330
USA
Tel + 1 914 937 4302
Fax +1 914 937 2485

Asia/Pacific and Worldwide Headquarters
Ideas International Limited
Level 3
20 George Street
Hornsby, NSW, 2077
Australia
Tel +61 2 9472 7777
Fax +61 2 9472 7788

Europe, Middle East, Africa
Ideas International Europe
1 Deanes Close
Steventon
Oxon OX13 6SZ
United Kingdom
Tel +44 (0) 1235 437 850
Fax +44 (0) 1235 437 851

www.ideasinternational.com