A WebSphere business integration solution

July 2002

WebSphere® software

Application connectivity lets you act now while you plan for the future.

Contents

Introduction

Our society has changed from a local to a global economy. This shift has had an impact on business and how it is conducted. Technology has become a crucial factor for business, to the extent that the term *e-business* is widely accepted. Consequently, it's easy to lose sight of what this economy can mean to a company and why it's so important. Today, a successful business needs to have a strong Web presence. At the core of e-business is technology that enables a user with a Web-enabled cellular phone, personal digital assistant (PDA) or Internet browser to access information and services from millions of Web sites. e-business taps into the potential of a world that has moved to a new era of connectivity. This white paper describes two aspects of connectivity impact: business and technology. It explains how your company can rapidly deploy technology now that can address the requirements it needs today to fully exploit e-business, and take advantage of emerging Internet standards as they become established.

e-business creates new opportunities

Major opportunities arise from e-business to help improve your company's performance. You can take advantage of the opportunity to:

- *Improve operational efficiency. Gain a larger return on your current investment in employees, business operations, applications and existing information technology (IT) infrastructure to help increase profitability.*
- *Consolidate. Execute mergers and acquisitions, while reducing the risk and cost involved through effective integration of separate systems and business processes. This lets you improve the benefits of synergy by eliminating redundant or duplicate functions.*
- *Improve customer service while helping to reduce costs. Exploit technology to create new opportunities to interact with customers, promoting loyalty and repeat business. This allows you to integrate fragmented customer information and streamline disjointed customer service activities.*
- *Collaborate with trading partners. Improve the efficiency of your supply chain to help reduce costs and better match supply with demand.*
- *Automate procurement. Help reduce purchase-order processing cost and time while improving business flexibility and agility. This allows you to devote more resources to other components of your business.*

- *Link with strategic trading partners. Help cut costs and gain flexibility by outsourcing noncore business operations. This lets you attract additional business by providing services to other firms.*
- *Integrate the virtual enterprise. Link business functions–from front-end customer relationship management (CRM) to back-office supply chain management (SCM). This helps you achieve straight-through processing from customer demand to product and service supply.*

e-business drives newer and better technologies

Technology provides improved access to information and the ability to easily share it. Collaborating around new ideas is possible through the connected world of the Internet. This has led to rapid convergence on open industry standards–for example, XML as a universal metadata notation. Recently, much debate has centered around higher-level standards. Most notable are Web services, which define how application services can be publicized, discovered and invoked to improve application interoperability. Also business protocols–like ebXML–address issues around languages organizations use to interact. The combinations of these standards help to drive down the total costs of integration, and make it progressively easier to deliver the typesof business projects already mentioned.

While discussion on open standards proceeds, more development needs to be done before a comprehensive set of standards that address all levels of connectivity–from networking to business interactions–is universally accepted. Even though e-business continues to evolve, an organization can’t afford to ignore the pressures or delay exploitation of the opportunities that e-business brings. Decision makers are faced with a dilemma: How to meet current business needs when implementing today’s technology and standards, yet keep options open for the future? IBM application connectivity can help you design and implement an infrastructure that allows business units to share information and quickly take advantage of new technologies and standards.

A strategy for e-business architecture

A vital part of e-business architecture is connectivity, but IT systems do not connect automatically. And when applications are developed under budget constraints and project deadlines to meet specific business goals, they often have a limited ability to interoperate with other applications. More sophisticated capabilities – that bring the ability to manage potentially unlimited permutations of operating systems, networks, programming languages, data formats and other implementation variables – lead to greater complexity, time and cost.

Industry standards are beginning to address these issues. For example, XML provides a standard method to mark up application data, so that communicating applications can identify individual data elements. TCP/IP is close to becoming the universal standard in networking. Web services standards are striving to provide a standard way for distributed services to be invoked.

A complete strategy for e-business architecture requires more than connectivity using industry standards. Applications need to be able to connect and exchange information as part of normal operation, which requires flexibility to handle continual change. These applications must also be robust to overcome the unpredictable demands of the real world. The following sections discuss some key capabilities for information exchange.

Routing

The destination for information should be independent of network topology so that routing can be separated from application. This anticipates network outages and delays so that the requirements of applications for timely delivery of information and the many possible types of distribution are met: one-to-one, one-to-many and one-to-anonymous, when the sender is not responsible for selecting the recipients. Networking technology is evolving to support these modes of operation. For example, an Internet Protocol (IP) multicast can deliver a packet of data simultaneously to multiple destinations supported by network routers. But without access to the structure and meaning of data within packets, network function is limited in how far it can go to support application-specific requirements.

Store and forward

An exchange between applications can be interactive. For example, a request is made and the response is awaited. A lost or delayed response can be handled in the application by trying to resend the request or taking an alternative action. However, an exchange may be one-way: that is, an application sends an update and does not expect a response or it can allow a time delay before expecting a response. In this case, more logic is required in the sender's application to ensure the receiver actually gets the information. Storing the information external to the applications can make the design of the applications much simpler.

Integrity

Many points of failure can occur in a complex technology environment. For example, power can be lost, hardware may fail, software could contain a bug or the network might become overloaded. Designing software to guard against failure and maintain integrity of information can be a major challenge. Database systems provide features that allow an application designer to easily maintain information integrity. Integrity can be regained, for example, following an unexpected failure by coordinating updates to the database. A deletion on one side of the system is matched by an addition on the other side – either both happen or neither happens. In the same way, when information is being transported, it is important to guard against unexpected failures. Enabling information to be moved from a database to a message as it travels from one system to another and then added to another database without being lost or duplicated – even during a system failure – requires sophisticated logic. Providing this logic, so that the application developer doesn't have to create it, makes applications easier, faster and less expensive to develop – and more reliable as they run.

A managed health care provider needs to integrate and consolidate an acquired company's applications and have access to its information and data.

Because information has significant value to e-business, there is a need to share it between applications. But not all information is equally valuable. Certain types of information are unique. For example, a request to purchase a number of shares of a company stock. Any loss or duplication of this information would have dire financial implications. This type of information requires integrity. Other types of information, like the result of a query on a bank balance, can be duplicated by a system or even lost without negative repercussions. The system can simply request the information again without a user becoming aware.

An international stock exchange needs to provide superior customer service through multichannel integration, and present an integrated customer view on a high-performing, reliable Web site that is scaled to volume—without higher costs.

Enabling integrity is more complex than simply propagating a message, which means it can be slower. The trade-off between integrity and speed is one that the integration designer must make. But with middleware, you can have both options. The choice is implemented by simply specifying a parameter. Certain information is also time sensitive. For example, with a request to purchase shares, the price of shares is volatile and the message should be processed as rapidly as possible, while preserving integrity. Enabling the application to specify the relative importance of information allows the middleware to set priorities for the processing of messages, so that important or time-sensitive messages are processed first.

Robustness and reliability

Valuable information that must be processed in a timely fashion leads to the requirement for resilience in the infrastructure. In the event of a catastrophic failure of one piece of the solution, there must be redundancy so that an equivalent piece can assume responsibility for handling the exchange of information. Where possible, the takeover should be performed automatically and swiftly. Redundancy implies replication, which is an advantage when systems are available because it provides greater capacity. It also presents challenges when you coordinate work across replicas to manage affinity where multiple, related parts must be handled by the same piece of the solution. This allows a multiple-step process to be fed by a complete, multiple- part stream of information.

Scalability

While replication is one approach to providing greater throughput, there are other aspects to consider as well. Performance involves getting maximum results from the resources employed, whereas scalability is how the volume of required work is handled. A higher performance is achieved with an optimized design that takes advantage of shortcuts to deliver the functionality required with least effort from the system. The first aspect of scalability, throughput – handling large volumes of data at fast processing rates , is achieved through performance and capacity. The other aspect of scalability is handling large sets of data. Two large data sets are relevant – the first is data being processed by the system and the other is data controlling the behavior of the system.

Because processing large amounts of data by the system requires large amounts of resources—like memory and disk space—the middleware must be able to break down the data into more manageable chunks that can be handled within the constraints of the environment. This requires capabilities in the middleware to support applications and its own internal operations. Where large data sets contain information about the system behavior, such as configuration information, the middleware must be able to operate efficiently to maintain performance. One example is a publish and subscribe broker, where thousands of subscriptions stored by the broker must be evaluated against each message.

Security

According to a recent report, when it comes to e-business, controlling and managing security is a primary concern of IT managers[1]. With wide-reaching connectivity comes the potential for information to be intercepted, read and even modified or misrepresented. Security is another area where standards and technology are constantly evolving. As new techniques are continually being developed to stay ahead of increasing computing power that threatens to overcome mathematical challenges. To allow flexibility in implementing appropriate security features, the middleware should support plug-in points where security functions can be inserted.

Security applies to application connectivity in three ways: First, it means controlling access to the configuration of the middleware by verifying authorized privileges against user identity. Second, it involves verifying authorization for applications to use the services of the middleware. Third, it may require authentication of information so that a recipient can verify that the information received is intact. In addition, the information may be encrypted to make it unreadable to any nonauthorized recipient. Finally, a combination of application identifier and authentication or encryption can be employed to enable nonrepudiation, where the recipient can accredit the identity of an originator.

Federation

Overall control of a middleware solution includes the implications of a far-reaching infrastructure, with shared ownership between multiple departments and across multiple organizations. Requirements exist for the appropriate span of control to limit the parts of the overall solution each use may affect. This control includes configuration, definition of security and authorization, availability and accountability through logs and audit reports.

Application programming interfaces (APIs)

An application invokes the services of middleware through an interface, which is usually manifested as a set of function calls that the application developer includes in the code. The set of functions and the associated data structures are known as an API. The interface should be standard across multiple languages, such as C and Java™ technology-based components, and environments like IBM CICS® and IBM IMS® to promote portable design, code and skills.

To ease the task of the application developer, the API should be simple and intuitive to use, with basic functionality achieved with few functions and simple data constructs. More sophisticated features should be accessible through the same interface and provided as extensions to basic functions rather than rewrites. As the application and middleware are updated, the API should preserve upward compatibility so that newer versions of middleware do not require rewrites of the application code to preserve existing behavior.

Challenging IT environments

Beyond the traditional environments of desktop PCs, machine room servers and fixed networks is a rapidly expanding universe of mobile and pervasive computing devices. These devices are connected to each other and to traditional systems through wireless networks that deliver highly variable service. Designed to provide specific functionality, the functions required from middleware are a subset of the list described so far. They must be completely embedded within the application, so that the user only sees the features provided by the application and is unaware of the middleware underpinning the application. For example, a user wants to be sure that orders entered on a PDA are reliably transmitted to the sales order application at the head office, without worrying about multiple configuration options for a middleware solution. The limited resources of these devices present even greater challenges to performance, capacity and robustness for traditional systems. To support the exchange of information between applications, a great deal of functionality lies between the logic within an application and the basic transport of a network. Middleware is designed to provide this functionality and can be thought of as a layer above the network that provides capabilities to simplify the design and implementation of application logic.

A major automobile components maker needs to create a single infrastructure to support and connect different applications on various systems, using a single interface to customers.

Implementation strategies

Middleware is usually implemented in an organization on a project-by-project basis. It's rare – particularly in the current economy – to realize the extraordinary changes in business value from technology and investment in strategic infrastructure without specific objectives for costs and benefits. At the same time, making buying decisions on a project-by-project basis usually leads to extra costs because there are additional integration problems where systems with different designs and technologies interoperate. Middleware technology is capable of addressing a wide range of integration needs from application connectivity functions to process integration.

Once information is unlocked from applications, it also becomes possible to deliver it to users in new ways, with enhanced structure and dynamic content selection through business rules and inference through portals. A solution that can be deployed on a project-by-project basis, but has the functionality, flexibility and scalability to support many projects, can gradually become a consistent enterprise-wide infrastructure.

Another recent industry report from Gartner shows that when calculating the ROI of application integration, companies are not only concerned with the savings on software and IT, but also with the returns on improved business processes across the organization and beyond.[2] As the importance of IT goes from being one of many factors in improving operational efficiency to being the single key to increased flexibility and business effectiveness, the need to control IT costs while delivering business benefits is growing. You can control costs of IT in a number of ways. Progressively reducing development efforts required by IT resources increases the value of existing assets through reusable interfaces. Reducing redundancy in the IT infrastructure by cataloging equivalent function and then progressively eliminating duplication is also an effective method. And you can speed time to market by minimizing the impact on other systems.

Selecting a consistent infrastructure offers ongoing benefits that can help lower cost of operation because management and control of the infrastructure is simpler than a multivendor solution. Resolving problems between multiple vendors' products, for example, can consume a lot of time and energy. It's often easier for designers and developers to work with product documentation from a single vendor with a consistent style and approach. Similarly, education needs can be met with courses and learning material that are consistent and modular. Other learning opportunities, like conferences, cover multiple products in depth from the same vendor, so attendees gain more benefit from attending. For example, *developerWorks Live!* is a conference that covers IBM WebSphere® software products and other IBM software.

Salary is the largest expense for most IT operations. Building broad employee skills that are reusable in multiple projects can bring significant long-term cost savings. Purchasing software from a single vendor can be more cost-effective with less procurement process than dealing with multiple suppliers, more flexible with generic capacity units and more discounted by reaching higher-volume levels. Cost-effectiveness at every stage of deployment is important because greater return on investment (ROI) in the short term can lead to more profit that can be invested in long-term future enhancements.

Standards adoption

An international tire manufacturer needs better business-to-business (B2B) communication with dealers, logistics partners and automobile manufacturers must adopt integrated business processes to bring together separate application systems.

Standards are important because they lead to improved compatibility and interoperation. They offer more choice, which provides greater flexibility and drives down costs. And because standards are developed through collaboration by groups of interested parties, they represent a consensus of opinion among industry players. When vendors adopt a standard, less risk exists for customers that a vendor's technology will radically diverge from the rest of the industry. This means that current investment is better safeguarded against being left tied to a dead-end technology. Solutions that are exclusively standards-focused often bring an unnecessary burden of having to bridge to heritage environments that are not standards-compliant. Ideally, the solution will be inherently open without being limited to standards. Being able to support fundamental standards like XML, as well as proprietary and legacy approaches, is key to providing this flexibility.

For example, in this scenario you would be able to take an application that doesn't support XML or Web services and make it accessible to architecture designed around services that are described, publicized and invoked through other open standards like Web Services Description Language (WSDL), Universal Description, Discovery and Integration (UDDI) and Simple Object Access Protocol (SOAP). The reverse would also apply, in which a legacy application provides a service to others through traditional mechanisms, replacing the application with a new, Web services-enabled solution. This requires that other applications that continue to use the service can have their traditional invocations mapped into the Web services equivalent.

The advantages of buying compared to building

Why choose a middleware solution? The decision to buy middleware rather than building a solution is a difficult one. In a recent IBM survey, nearly 70 percent of IT decision makers said they use custom programming for integration.[3] Many of these decisions are based on solving incremental problems that began when a new application was designed to connect to one or two existing applications.

On this small scale, without considering future requirements, the choice of writing some simple code using basic networking facilities seems straightforward because the effort to develop and maintain this code is probably no more than using a middleware product. However, if the requirements change, even during the design phase of an application development project, the additional complexities rapidly skew the costs. Middleware code has functionality to guarantee integrity, provide flexible routing and manipulate application data in transit, which requires little or no extra application coding. Developing equivalent function in the application quickly becomes very complex and therefore costly to develop and maintain.

Middleware vendors, on the other hand, have the benefit of recouping their development investment from a large number of customers. Compared to unique application code, middleware provides superior feature value. Moving responsibility for portions of the overall IT function out of application code and into middleware results in less responsibility and risk as you fix bugs in the software, keep current with prerequisite software — such as operating systems — and deliver the best performance from the available resources. Overall, it means lower total cost of ownership and this, in turn, leads to greater overall IT ROI.

Product	**Function**	**Benefit**
IBM WebSphere MQ	Messaging integration	Integrate applications across more than 35 platforms with once-only message delivery.
IBM WebSphere MQ Everyplace™	Mobile integration	Securely extend your e-business integration to a mobile workforce.
IBM MQSeries Adapter Offering	Application adapters	Rapidly develop adapters to integrate custom applications.
IBM WebSphere Adapters	Packaged application adapters	Utilize prebuilt application adapters to integrate packaged software, like Ariba Buyer and SAP.
IBM WebSphere MQ Integrator	Wide-scale messaging integration	Enable dynamic, distributed, event-driven application integration.
IBM WebSphere Data Interchange	Electronic Data Integration (EDI) translation	Translate business information for exchange with trading partners.
IBM CrossWorlds® Interchange Server	Process-based integration	Coordinate business applications across multiple processes.
IBM CrossWorlds Collaborations	Industry-specific solution templates	Leverage proven industry solutions for fast deployment.
IBM CrossWorlds Trading Partner Integration	Business-to-business integration	Automate supply-chain interaction through a gateway for business-to-business communication.
IBM WebSphere Partner Agreement Manager	Business-to-business integration with RosettaNet	Participate with trading partners in RosettaNet communications.
IBM MQSeries® Workflow	Business process management	Define and execute complete business processes.
HOLOSOFX BPM Suite	Business process modeling and monitoring	Model, simulate and monitor business processes that execute in the runtime for MQSeries Workflow.

Business integration building blocks

With the IBM WebSphere MQ family, you can combine business integration products and solutions selectively or build one upon the other to help integrate business processes within and across enterprises. WebSphere MQ application connectivity is part of the IBM WebSphere software platform for e-business—a set of integrated, award-winning e-business solutions. No matter where you are in the e-business cycle, the WebSphere software platform can allow you to grow—at the speed the market demands. Building on this robust platform, you can connect diverse IT environments to maximize your current investments and leverage existing skills with a full range of business integration solutions from the IBM WebSphere MQ family of products.

Summary

Application connectivity is a cornerstone for IT infrastructure in e-business. A comprehensive solution that delivers critical function in a modular and flexible way can help to deliver business results, both in lower costs and increased revenues. It also provides the basis for exploitation of new technologies and standards as they evolve. WebSphere software provides comprehensive application connectivity capabilities.

For more information

You can start using WebSphere MQ today on more than 35 platforms, across virtually all of the major networking systems, to connect applications, so they can exchange and leverage information for greater business value. Getting started with developingan application using WebSphere MQ requires only six basicprogramming functions that place you on the path of powerful capabilities that unfold as your needs dictate. To download WebSphere MQ software and documentation from the Web, visit **ibm.com**/websphere.

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

Produced in the United States of America
07-02
All Rights Reserved

[1] *CIO Magazine*, Tech Poll, March 2002.

[2] Roger Fulton, group vice president and worldwide director, Gartner Research & Advisory Services, Dataquest Software Management, May 2002.

[3] IBM VLE CIO buying survey, March 2002.

e business software