Exploiting application connectivity to enhance your e-business.

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 mobile 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 language. 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 cost of integration, and make it progressively easier to deliver the types of business projects just mentioned.

While discussion on open standards proceeds, more development needs to be done before a comprehensive set of standards that addresses 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 software can help you design and implement an infrastructure that allows business units to share information and quickly take advantage of new technologies and standards.

Business integration has been around for a several years (for example, with Electronic Data Interchange [EDI]); however, the need to accommodate e-business is driving it in new directions. Enterprises are caught between exposing their applications to customers in a simple, unified manner (for example, using a portal) and interacting with other organizations in a responsive, more cost-effective way (for example, business-to-business [B2B]). There is pressure to do more with less internally, which is forcing a move to new and better ways to increase productivity. In all cases, one thing is clear, business integration, through the use of middleware and its tools, offers a huge potential to help solve these issues. Whether it's to increase return on investment (ROI), to achieve faster time to market for new business function or to lower the risk of deploying that function, integration can play a vital role.

To address these issues, business integration delivers five key capabilities that customers have recognized as critical components to achieve effective, successful integration solutions across an IT infrastructure. Whether deployed separately or together, they are designed to meet a comprehensive set of business needs. Specifically, these five capabilities — model, integrate, connect, monitor and manage — are necessary to help customers turn business innovation into business results:

- *Model to design, simulate and plan business process. Business professionals can participate in innovative business process design and help model current and plan future processes.*
- *Integrate to link people, processes, applications, systems and data. Leverage new best-of-breed applications and legacy systems through real-time business process integration.*
- *Connect to extend those processes to customers and business partners. Accelerate e-business initiatives and deliver business agility to react quickly to market changes in value chains.*
- *Monitor to control and track business process end to end. Observe and track the state of business process events and take corrective action to improve business execution without affecting IT infrastructure.*
- *Manage to review and analyze business effectiveness and improve overall business performance. Analyze and improve business process operations.*

The integrate capability is provided by WebSphere® business integration software from IBM. It enables agility and flexibility within the enterprise, allowing customers to grow with their business needs. It does this through two very important styles of integration: process integration and application connectivity. Built on market-leading experience in integration and open industry standards, WebSphere business integration allows you to harness the full potential of your company. WebSphere business integration is a family of products that includes ready-to-use solutions for common business problems, as well as industry-specific offerings and an easy way to extend your enterprise to include your partners. WebSphere business integration will help you accelerate and simplify your e-business initiatives with secure and scalable technology that leverages your existing application and legacy investments.

A strategy for e-business architecture

Within the integrate capability, 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 application-connectivity style of integration provides a standard means for applications to

connect and reliably communicate. Its role is to handle incoming events by routing them to the appropriate application or process, to provide the means for applications involved in a process to communicate in a request/response manner and finally, to handle the disposition of any messages that may result from the completion of a process. Application connectivity can be viewed as a combination of a high-speed reliable transport with the ability to connect to other transports and an in-flight messages translator that reconciles the data difference between connecting applications. Specifically, the functions provided by application connectivity are: message transport, event brokering and message brokering.

The following sections discuss some key capabilities for this type of information exchange.

Message transport

Message transport provides applications with reliable, point-to-point messaging. This simplifies applications by moving much of the complexity needed with a distributed design into the middleware. It also permits applications to be loosely coupled, which allows applications to appear as if they are always available. This provides a flexible means to temporarily defer work (for example, to batch orders for a nightly run or during the period when an application is serviced for maintenance). Message transport is appropriate for complex designs found in large organizations, as well as for simpler designs found in small and medium businesses. It is especially useful for applications that are initially targeted for smaller environments, but eventually are used in larger complex environments.
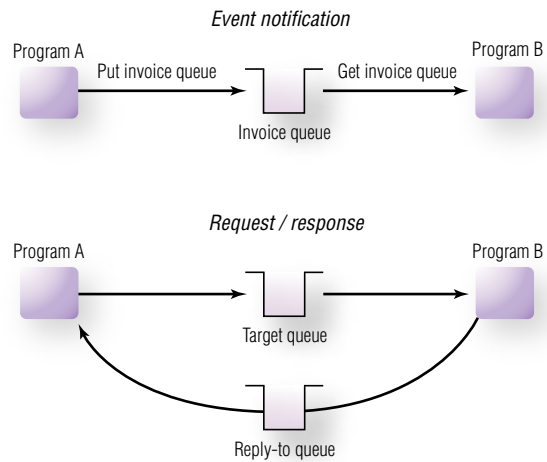
Messaging styles

A message exchange between applications can be one-way or two-way. In a one-way communication, a message is sent but no response is expected. Typically, this happens when one application needs to notify another that an event has taken place. In contrast, in a two-way communication a message is sent and the recipient returns another message. This pair of messages forms a request/reponse. In both cases, the applications do not directly interact with each other. Instead, they interact with middleware that provides the message transport. Applications read and write messages to a queue in a manner

similar to the way they read and write records to a file. So it is the middleware that assumes the responsibility to deliver messages in a store and forward manner that can be immediate or deferred, depending on the availability of the applications.

Messaging styles

Event notification

Program A   Put invoice queue   Get invoice queue   Program B

Invoice queue

Request / response

Program A   Program B

Target queue

Reply-to queue

Simple application connectivity using message transport

Event brokering

Event brokering extends the message transport capability by providing applications with the ability to exchange messages based on a publish and subscribe model. This is particularly useful in environments where applications or their interaction with messages are particularly dynamic. For example, one application may respond to process all order messages while another responds only to orders exceeding US$10,000. In addition, event brokering extends the message transport capability to applications that communicate over other protocols (for example, those applications using IBM WebSphere Everyplace® and Supervisory Control and Data Acquisition [SCADA]).

Event brokering
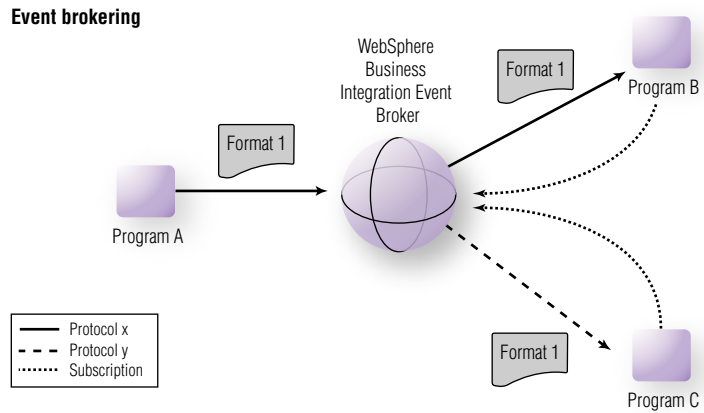
Extending message transport with a publish and subscribe model

Routing

The destination for information should be independent of network topology so that routing can be separated from applications. 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.

Message brokering

Message brokering extends facilities of the event brokering by adding the capability to transform messages. Messages can be enhanced through the addition, reordering, replacement or deletion of fields. If needed, data for new or replaced fields can be obtained through database or table look-ups. Using message brokering further simplifies application design since the responsibility of transformation is moved out of the application and into the middleware.
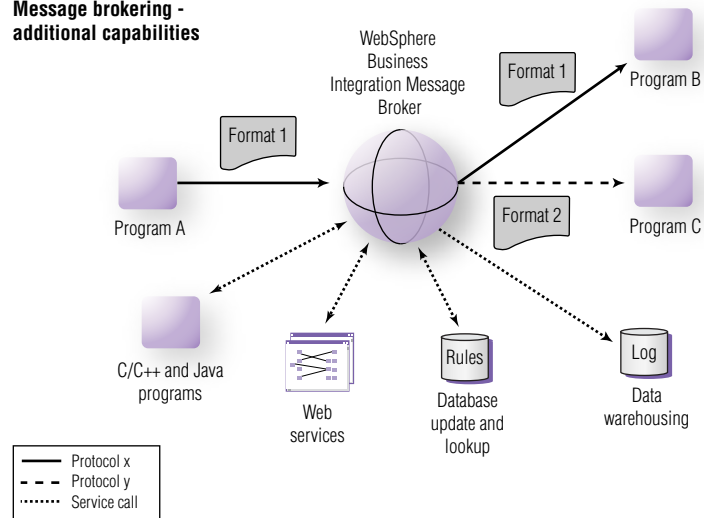
Message brokering - transformation
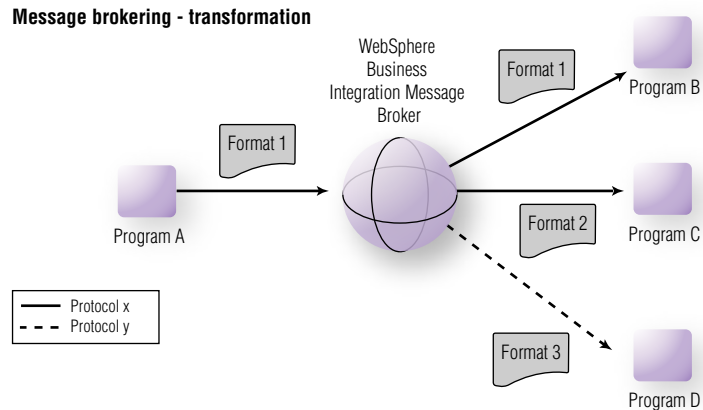
Message brokering enables message transformation to multiple formats, across multiple protocols

Because brokers operate on messages as they flow between applications, there are two other highly valuable functions they can provide, namely, to expose an application as a Web service and to warehouse data. In both cases, this is done without the need to modify or redeploy the application. In the first case, a Web service is advertised with the broker named as the service provider. When a request is received, it is turned into an ordinary message and delivered to the application. In turn, any response message generated by the application is converted by the broker to a Web service response. The application processes this as a message, not a Web service. This broker capability allows organizations to rapidly adapt legacy applications to this emerging, standards-based technology.

In the second case, the broker updates a database as part of its processing of a message. Whether a given message triggers an update is determined by a set of rules that can accommodate a number of different criteria (for example, a message's originator, destination and content). Through the use of this broker feature, corporate or government policies that require monitoring and reporting can be rapidly deployed, again without the need to change any applications.

**Message brokering -
additional capabilities**

WebSphere
Business
Integration Message
Broker

Format 1

Program B

Format 1

Program A

Format 2

Program C

C/C++ and Java
programs

Web
services

Rules

Database
update and
lookup

Log

Data
warehousing

——	Protocol x
– – –	Protocol y
········	Service call

Message brokering with WebSphere technology enables important additional functions

Transformation

Often, when a routing decision results in a message being sent to a new or additional application, it must be transformed so that it is consumable by that newly targeted party. Typically, this implies one or more of the following alterations to the message:

- *Add fields. For example, include additional information on a diagnosis (county of residence in addition to city and state)*
- *Delete fields. For example, eliminate data for reasons of privacy (a person's name on a medical diagnostic message)*
- *Modify fields. For example, convert customer numbers from one company's form to another as a result of a merger*
- *Reorder fields. For example, exchange "first name last name" with "last name, first name"*

Characteristics of integration

It is easy to understand the compelling reasons for business integration. WebSphere business integration is specifically designed to allow companies to start small and grow. Therefore, it is possible, and often desirable, to begin with message transport, then to add event brokering and eventually expand to

message brokering as a messaging backbone takes hold in an organization. Process integration, too, can be added at any point to meet the integration needs that require a more complex interaction of applications, thus completing the integrate capability. In some cases, it may be desirable for one of the other capabilities to be introduced at an early stage (for instance, model) in the evolution.

WebSphere business integration provides a highly flexible approach to the adoption of integration technology because it is often not practical or affordable to implement all the technologies at one time. The following section describes the key characteristics of integration.

- *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.

  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 the user becoming aware of the problem.

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 involves the way 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 in this scenario — 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*
  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. 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 can 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.

A single way for application-connectivity development

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 application programming interface (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® systems, 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.

Connecting diverse, distributed environments

Beyond the traditional environments of desktop PCs, machine room servers and fixed networks are 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. 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.

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 and can gradually become a consistent enterprise-wide infrastructure.

Application connectivity is one aspect of business process management (BPM) technology, which can save time, money and add value to your IT infrastructure. It also spans the business and technological gap to create synergy, with proven results.[1] As the importance of IT changes 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 in a number of ways. Progressively reducing development efforts required by your 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 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, IBM developerWorks™ Live! is a conference that covers 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 ROI in the short term can lead to more profit that can be invested in long-term future enhancements.

Standards adoption

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 your 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 type of solution you can 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. 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[2]. 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 available resources. Overall, it means lower total cost of ownership and this, in turn, leads to greater overall IT ROI.

Meet demand with market-leading business integration solutions

Integration: the efficient, flexible combination of resources that enables you to optimize operations across and beyond the walls of your business. By tying together disparate resources, data, applications and processes, you can create a single, cooperative environment to unite your entire demand chain. And speed your return on investment and time to value.

With WebSphere business integration solutions from IBM, you can connect:

- *People. By integrating business processes, your customers, employees, partners and suppliers can interact with your business more efficiently.*
- *Processes. By managing and coordinating your entire business horizontally, your IT infrastructure can keep pace with demand—and your business objectives.*
- *Information. By extending real-time access to distributed information, people and processes can take advantage of data and content resources from a variety of repositories.*

In your transformation to e-business on demand™, you'll need to increase efficiencies. Streamline processes. Leverage existing assets. And provide immediate, dynamic, personalized information that helps build long-term relationships. The WebSphere business integration software portfolio delivers.

Built on open, industry standards like Java 2 Platform, Enterprise Edition (J2EE), XML and new Web services standards, WebSphere business integration software helps your business become more focused. Flexible. Resilient. And better able to respond to the opportunities — and disruptions — that can influence your future growth.

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 from IBM provides comprehensive application-connectivity capabilities so that your business can adapt and thrive, regardless of how technology evolves.

For more information

To find out more about application connectivity and business integration, visit **ibm.com**/software/integration.

[1] Jim Sinur. Gartner Research and Advisory. The Business Process Management Scenario. June 2003.

[2] IBM VLE CIO buying survey. March 2002.

business software

G325-2038-01