A Road Map to eBusiness Infrastructure Integration Challenges

An IDC Brief

Adapted from: Enterprise Integration Software: Platforms, Middleware, and Applications for eBusiness (IDC #24216, March 2001)

The evolution of ebusiness to what many in the industry are now calling Web services is accompanied by new approaches, standards, and enabling technologies that have made possible the implementation of new architectures and infrastructures that support the rapid development and deployment of ebusiness applications. IT managers face significant challenges regarding the increasing complexity of integration and the skill sets needed to implement the new ebusiness infrastructure. Today, ebusiness is a competitive imperative, and the current economic downturn should not dissuade organizations from moving forward in their ebusiness efforts; the competition isn't waiting.

A few tools and methodologies available today build on experience and knowledge accumulated over the relatively short time over which new ebusiness infrastructures have been designed and deployed. However, no "cookie cutter" approach exists to making choices regarding infrastructure integration. One thing is for sure: The metrics by which the efforts to build these infrastructures will be measured — based in tried-and-true concepts such as customer satisfaction and responsiveness — still form the basis for success.

IDC
Analyze the Future

The Evolution of eBusiness

From "Baby Steps" to Web Services

The first steps organizations took to leverage ebusiness concepts were based on the notion of "getting to the Web." Most organizations' early efforts were spent building Web pages to provide information and some support for business transactions. However, implementations reflected a relatively simplistic view of existing internal applications — one, perhaps two, "stovepiped" applications (meaning applications designed to run alone) were brought out to the Web, and seldom were these applications integrated with each other. Furthermore, processes that supported Web transactions were often not linked to them or automated in any significant way, resulting in unreliable support in terms of time and completeness. Finally, the tools used for the development and deployment of Web-based applications were immature, making development difficult by those who were as yet unskilled in Web-based development.

While many businesses have stopped at this initial phase of ebusiness, ebusiness capabilities have now evolved to a level of sophistication that makes both higher levels of application integration and business-to-business (B2B) interactions possible and often mandatory. New enabling technologies such as XML, and integration paradigms that embrace business process modeling and integration have made possible the integration of applications and business processes both across the enterprise and among enterprises. This has enabled the creation of a coherent view of all business interactions, thus making possible a high level of support for automated business processes.

The next step in the evolution of ebusiness has just begun — the move toward what IDC calls *services-based integration*. Services-based integration acknowledges the fact that:

- Application functionality will soon be represented as a service.
- Services may reside anywhere on a network.
- The precise location and means of access of a service may not be known.
- A mechanism to identify and locate a service must be provided.
- A growing variety of access devices and modes necessitates an infrastructure that will allow the bidirectional flow of information, transactions, and so on between those devices and the network and services-based infrastructures with which they must interact.

Although the definition of Web services is still evolving, the functionality embraced by services-based integration reflects in great measure what Web services purport to be.

While Web services implementations are few and far between today, it is likely that over the next few years a good portion of the efforts of IT organizations will be focused on understanding the associated concepts and on beginning the process of building solutions that reflect these concepts. The increasing complexity associated with the evolution of ebusiness will continue, and most likely accelerate, as Web-services–based solutions appear. The following key challenges to IT managers are all factors that will affect the ability to achieve success in ebusiness:

- Integration of existing assets (applications, business processes, data, information, and content)
- Security
- Personalization that is intertwined with secure, focused, and controlled access to information
- Skills required to build and support these integration-focused solutions

Several truths must be accepted:

- eBusiness is an imperative. Regardless of the degree to which businesses manage themselves traditionally, an ebusiness presence — tailored to the specific business model of the organization — has become a strong component of the competitive mix.
- The primary goal of ebusiness initiatives should be enhancing business opportunities (and efficiencies associated directly with delivering on them) for additional competitive advantage. Cost cutting and efficiencies should be a secondary goal. Success won't be about who does it, but rather about who does it better.
- The most important technological challenge facing IT organizations today is integration. The solutions required will reflect the need to integrate at a variety of different levels — from applications and data, to business processes, and finally across (and within) organizations and in a way that embraces all possible sources of complexity. This challenge is forcing the creation and adoption of new infrastructures and enabling technologies that will be used to facilitate this integration.

What About xSPs?

IDC has created a term to describe the growing variety of service providers: xSP. An xSP provides services for delivery over a network for a fee. These can include application service providers (ASPs), network service providers (NSPs), development environment service providers (DESPs), system infrastructure service providers (SISPs), and others. The purpose here is not to create a new age version of "alphabet soup," but to accentuate the commonality of these types of organizations relative to the infrastructures needed to support their businesses.

In fact, the concepts, challenges, and proposed actions presented here apply to xSPs as equally as they do to the IT organizations within corporations of varying sizes. Despite the fact that the business model for an xSP is founded in the notion of "one-to-many" delivery (which may not be true for an internal IT organization), the architectural and technological requirements, as well as the metrics for the success, are similar and, in some cases, identical. As the Web services model becomes more prevalent, IDC expects that internal IT organizations will be forced to behave more like xSPs, responding to the needs of their constituents (both internal and external) in the same ways an xSP must respond to its customers.

Architectures for Effective eBusiness

The architecture required today has to satisfy some very specific needs relative to the Web and, looking toward the future, Web services, which require the following characteristics:

- **Multitier architectures.** Because a wide base of legacy applications represent numerous technologies, languages, and standards, it is important to separate the mechanism by which those applications are integrated with the Web and with each other. Architectures that accomplish this usually encompass three or more "tiers" — levels of functionality implemented in hardware, software, or both — that accomplish a specific set of goals. Existing applications and data generally make up one tier, access devices another, and the one or more tiers that reside between them make up the "middle tier." Application servers, Web servers, data caching and persistence functions, and other capabilities bring important integration power that does not have a technological dependence on the other two tiers. This is in contrast to two-tier client/server implementations that do not

effectively separate applications and data from the mechanisms that tie them to users or other applications.

- **Standards-based technologies.** The potential for a complex environment that includes a variety of platforms, access devices, operating systems, languages, tools, and networks (a scenario that is further accentuated with Web services) requires support of standards-based technologies that create a one-to-many model for development as well as a "bridge" among disparate sources of data, information, and content.
- **Security, personalization, and directory services.** The breadth of access and interaction representative of ebusiness solutions, and especially those based on a Web services model, requires the ability to provide controlled and focused access by customers, employees, suppliers, and, in some cases, applications that can interact directly with each other without human intervention. Security, personalization, and directory services all play a role here (see also IDC Executive Brief *Strengthening End-to-End eBusiness Security and Privacy*).

Architectural Approaches

Initially, organizations used application servers as a foundation in their efforts to build solutions based on a new type of architecture. These products are basically execution and deployment environments for component-based solutions that add management services (such as load balancing, failover, and, in some cases, data persistence and/or caching) that help distinguish it from other distributed solutions.

Application servers reside architecturally and logically on the middle tier, between backend systems that support existing applications and data sources and access devices (PCs, personal digital assistants [PDAs], automated teller machines [ATMs], and, more recently, mobile and wireless devices) on the front end. Mobile and wireless devices are of particular interest these days, both because of their increasing numbers and expanding variety, and because serving them requires some unique technologies for formatting, presentation, and connection.

The middle tier is logical and can actually be several tiers. For example, a Web-based architecture may support a Web server (supporting Java Server Pages [JSP]-based Web page services), application servers, and data servers for caching data for improved performance. The important fact to note is that no matter how many physical tiers make up the overall middle tier, all of them must provide the capabilities listed in the previous section of this paper.

Over the past two years, application server vendors, in response to customer demands for complete solutions, have expanded their product offerings to include enhanced integration capabilities, support for a variety of application-agnostic functions (such as personalization, portals, and security), and applications — manifesting themselves in what IDC calls ebusiness platforms.

While this distributed computing model offers flexible support for Web-based solutions, it carries with it a legacy of its own. Management functionality notwithstanding, business logic is still for the most part tied to specific systems and servers. Client devices are essentially "hardwired" to these systems, with access provided primarily through browsers or directly across a corporate network. Here, the term hardwiring describes the precisely defined connection between an access device or application and the systems and applications it accesses. Even a wireless device can be hardwired in this way. The Web services concept can, and will, change this.

Web Services

In a Web-services–based scenario, this hardwiring may not exist. One envisions a world in which knowledge of where specific business functions and/or applications (services) reside, the specific means of access, and the display and hosting characteristics of client devices are all an integral part of the environment itself. In essence, these hard wires are replaced with a set of services that can provide any mix of functions to any device from any place at any time. Keeping this access focused and secure becomes a major challenge.

The architectures to support such a paradigm require augmentation of the basic multitier version. The requirement now is for a comprehensive, network-based, ebusiness infrastructure that incorporates all the functions required to achieve the successful services-based integration. These architectures, which can vary depending on the specific purposes served, share some common characteristics:

- They incorporate a *services infrastructure*, which contains the assets (storage, applications, databases, etc.), platforms (application servers, ebusiness platforms, and support for mobile computing), and support functions (transaction support, security, directory services, etc.). This infrastructure is network-based, tying together systems and assets, sometimes in real time and as needed, to satisfy a particular usage scenario.

- They also incorporate an *access infrastructure*, which leverages the Internet to allow access to services by a wide array of access points and devices.

The two infrastructures are linked through an *Internet gateway* that handles the processing of transactions, data, information, and content so that it is contextually usable and understandable in and by both of them. This "edge processing" ensures that, as an example, the results of a transaction can be displayed in a readable form on a cellular phone or PDA display, and that information from a variety of sources is brought together to form a coherent picture of a business process, both within and external to an organization.

eBusiness Integration: The Challenges

Naturally, the comprehensive, standards-based architectural specification described above poses some unique and, in some cases, severe, but actionable challenges to IT organizations. How these challenges are met can mean the difference between success and failure.

Integrating the Legacy

While new business logic is often built using a component-based paradigm, legacy applications will be around for some time. The challenges include achieving integration without having to significantly change existing applications; solving the issues of handling transaction-based data; dealing with issues of performance through leveraging middle-tier services; and using standards-based technologies to aggregate information and data from legacy sources into a coherent, customized view of an application scenario.

Action: Organizations must perform a thorough assessment of the breadth, number, and type of legacy functionality they must integrate, in order to both scope the legacy integration task and build a foundation for making detailed decisions regarding the choices of architectures and enabling technologies to be employed.

Dealing with Unpredictable Demand

eBusiness often involves activities that make predicting online activity difficult or even impossible. Busy Web sites can often get loaded down with unexpectedly high user loads, resulting in poor performance, prematurely terminated sessions, and, ultimately, lost business. The challenge here is to combine as precise a business plan as possible with technological solutions that can handle unplanned load variations.

Application server vendors provide load-balancing technologies that can bring additional processors online when needed and balance loads among processors that are online at any given point in time. Some offer licensing schemes that provide a "time used per processor" option.

Other ways to deal with this problem include the middle-tier caching of Web pages (lowering the network demand generated to reaccess a page for which high demand exists), clustering (making potentially large amounts of middle-tier computing power available on demand), and data persistence mechanisms.

Action: Organizations must take a careful look at the purposing of their ebusiness solutions in order to assess the degree to which unpredictable load demands will be a factor, and they should determine which approach will yield the best payoff in terms of performance, reliability, and, therefore, to customer/employee/business partner satisfaction. Despite the technical nature of the solution, this is a business-focused, bottom-line issue.

Adapting the Organization for Infrastructure Integration

Because of the increasing complexity of ebusiness solutions and the environments in which they operate, a growing need for tools and technologies will allow for the abstraction of activities involved in building and deploying them. The paradigm for building integrated ebusiness solutions is shifting toward one that makes heavy use of visual modeling as an abstraction of functionality, such as assembly of business processes. This last activity, because it operates at such a high level of abstraction, can often be performed in part by nonprogrammers (typically, business analysts) who may, in fact, have little or no development experience.

Still, despite the fact that many of these tools automate the interfacing of business process steps with the applications and data on which they depend, some level of developer involvement will always be required to complete development and deployment tasks. The merging of efforts between business and IT personnel creates interesting and challenging organizational issues.

Action: IT organizations must make two major choices:

- **Choice of tools.** The first involves the choice of tools to be used to build business-process–based ebusiness applications. Traditional issues, such as the number of participants, their division by teams, and the geographic dispersion they represent, must all be taken into consideration. The expertise of programmers and business analysts must be assessed to

determine the tradeoff between tool sophistication and cost. The ability of tools to integrate with existing development tools must be weighed against existing (and predicted) tool investments.

- **Choosing organizational changes and goals.** The second decision involves a consideration of organizational issues. In most cases, business analysts and developers reside in different organizations, but with this new paradigm for building applications, they must work together toward a common goal while at the same time speaking different languages. Business analysts, for instance, may understand and be able to specify business processes required, but will they and developers be able to work together to work on compromises when schedule and/or cost issues require it? How will goals be set for both participant types to encourage such compromise?

Mastering New Enabling Technologies

The ability to build and deploy ebusiness applications today and in the foreseeable future will depend in great part on the implementation of infrastructures that can successfully support the Web services paradigm and a standards-based approach.

As an example, ideally a service should be available (and, therefore, deployable) on a variety of platforms (hardware, operating system, and application server combinations). This requirement acknowledges the fact that within and among organizations exists an advantage to be able to leverage all possible resources on an as-needed basis.

Web services need a more flexible one-to-many model that allows applications to be written once and run on any platform. Java, as an enabling technology, allows this to happen by creating a runtime and deployment environment that can be platform independent.

Another example is a reflection of the complexity of the integration task — the bringing together of data, information, and content from a potentially large number of diverse sources into a coherent collection (Web page, document, or even to support a specific business transaction or process step).

XML, as a meta language, makes it possible, in a customized way, to tag these assets and to transform them to a form that makes them understandable to one another and to the applications that use them. This is an extremely important and highly essential capability for building new ebusiness solutions, and its use is being expanded almost by the day. In recognition that assets can take virtually any form, XML is being applied to uses such as enabling directory services, to be integrated across

enterprises to provide comprehensive access control and security.

XML is also playing an important role in "edge integration," facilitating the ability to bidirectionally communicate between access devices and the services infrastructure.

Action: IT management must make appropriate choices of enabling technologies based on their organizations' system, software, and network profiles. IT management must also consider when a particular technology becomes enough of a "standard" to warrant attention. The importance of XML, for instance, is clear, but at what point do enough specifications exist to allow businesses with disparate applications and other IT assets to work together successfully and efficiently? The proliferation of standards is always at odds with competitive advantage for software vendors; careful thought must be given to the point at which a sufficient foundation is available.

Embracing Component-Based Development

For the past several years, software components have received a great deal of attention as an extension of the object-oriented development. The advantages include reuse, increased reliability, and ease of construction and assembly of entire applications. However, until very recently, adoption of component-based development has been slower than expected. In IDC's view, the primary reason for this has been cultural. Developers have traditionally embraced a view of their tasks that focuses on writing code specifically for use in new applications, and they have not been ready to work based on a model that leverages either reusability or code written by other developers. Yet the distributed nature of new ebusiness applications and the required component-based nature of the middle-tier platform of the ebusiness infrastructures being built today make component-based development a necessity.

While IDC believes that the adoption of Web services will serve as an accelerator for adoption of the component paradigm, the cultural issues must still be resolved. The catalysts for cultural change are straightforward.

Action: Top IT management must enthusiastically support the change and also provide goals and metrics that reward adoption of the component-based approach and make available a robust set of tools to support component-based development. These tools and repositories must be chosen to match the organizational structure and geographic profile of the development teams within the organization.

Shrinking IT Life Cycles to "Internet Time"

As ebusiness becomes a key competitive factor, shorter development cycles are required.

Component- and standards-based technologies can expedite the development process. In addition, application server vendors are adding to their ebusiness platforms elements of application frameworks that include application functionality in component form. These products provide a combination of collections of software components focusing on specific types of solutions; basic infrastructure elements to tie these components together and provide the interfaces to applications servers; other types of middleware, or operating systems; and the tools required to modify and assemble the components and tailor the infrastructure for a specific purpose.

Other factors also play a role in accelerating project lifestyles. These include the merger of the efforts of business analysts and developers and the use of tools that abstract the complexities of assembling business processes and automate their linkage to underlying applications.

Action: IT managers must make changes in their organizations and technology that help support the acceleration of project life cycles. These changes must reflect both the interdependency between the technologies chosen and the skills of IT and business professionals, and they must also reflect a change in the "development culture" within the organization.

Outsourcing Integration and Development

The next question is how to decide what to do inhouse and what to outsource. It is critically important for organizations to assess *early in the process* the parts of infrastructure integration that must be executed using outside resources. The assessment required is not an easy one, primarily because there is not a great deal of accumulated experience that can be applied by organizations to gain an a priori view of the return on investment (ROI) associated with various options for project execution.

Given the challenge of implementing new component-based technologies, instinct might prompt the mass training of large numbers of existing personnel at considerable cost. Another approach is to bring in experts in specific technologies that will transfer knowledge. The right decision will depend on backlogs, budgets, and timing — there is no "magic pill," but the factors to consider are clear.

For organizations embarking on the construction of new, multitier, services-based architectures as described earlier,

outsourcing may make a lot of sense, particularly to a system integrator.

Business models for system integrators are changing to leverage standards-based architectures that can be applied to a variety of client engagements, lowering overall revenue but still keeping margins high. This allows the extension of the reusability model to the overall architecture for infrastructure integration, while allowing a high degree of customization to the specific organization's needs.

System integrators can also help in the determination of an ROI for integration projects, since their history with a number of engagements gives them a base of experience and knowledge from which they can draw.

A likely scenario is for the overall architectural design to be outsourced to a systems integrator, with some of the more value-added aspects of the ebusiness applications to be developed internally. The degree to which external experts are brought in to assist in internal integration efforts will depend on the obvious but sometimes difficult-to-assess expertise with the tools and technologies required.

Action: Simply put, IT management must choose one (or a mixture) of the following obvious options:

- Execute on all integration tasks internally.
- Work on some aspects, and outsource others.
- Bring in experts for assistance and knowledge transfer.
- Outsource all aspects of the integration task.

Organizations must determine, in support of the option selection task above, which applications provide competitive advantage and, therefore, might be likely candidates for internal development**.**

Measuring Success

The final challenge is to incorporate into the infrastructure itself the means by which success can be measured. For instance, for a consumer Web site, a result of successful infrastructure integration might be the ability to achieve "stickiness" with customers. This can be measured by using technologies that monitor the site activities of specific customers, and by accumulating and analyzing statistics on site navigation and transaction completion. Another example is the use of software-based tools that can monitor overall system and network performance, providing insight as to how well usage surges are being handled, and even, in some cases, implementing analysis

methodologies that can predict load and performance problems as well as system failures in order to help increase overall reliability, and, ultimately, help achieve overall ebusiness success.

Action: IT managers as well as business managers must carefully consider the issues related to the challenges of measuring success from the standpoints of both client and business partner satisfaction and, ultimately, the organization's bottom line. In light of the specific situation an organization faces, effort spent here can help that organization move forward with the integration efforts required to create ebusiness success.

Summary: The Keys to Success

While paths to a modern ebusiness infrastructure may vary according to the individual company's needs and capabilities, how success can ultimately be measured is clear. Key success factors include:

- **High overall system reliability.** Is uptime consistent with demand? Have the correct technologies been brought to bear on the issue of dealing with system failures in a way that is transparent to all participants in the ebusiness activities?
- **High overall system responsiveness.** Is the performance provided consistent with capturing and retaining business? Are customers, suppliers, employees, and other participants being kept satisfied? Is your company measuring responsiveness in a way consistent with participant satisfaction, and is it measuring satisfaction itself in relevant ways?
- **Fast time to market of applications.** Are new applications being brought online, and are changes being made to existing applications, in a timely manner? Is your company responding adequately to a rapidly changing competitive environment?
- **Growth in knowledge to be applied to future integration projects.** Has your choice for executing on past and current integration projects (internal development, outsourcing, bringing in external expertise) resulted in a sufficient transfer of knowledge such that the ability to execute on future projects is enhanced and made speedier? Have you increased your ability to make decisions regarding how to execute on future integration projects?
- **Aligning the organization to a new culture.** Have you reorganized in a way consistent with new integration and development paradigms? Have you put in place goals and

metrics consistent with these paradigms? Have you successfully merged the efforts of business and IT personnel toward a common view of the construction, deployment, and maintenance of business-process–centric applications to support ebusiness?

- **Successful adoption of standards-based approaches to integration.** Have you implemented the right choice of standards in your solutions?

There are no surprises here — success will depend on the effective implementation of established, long-standing metrics for measuring how well business is executed. The trick is to do so within an ebusiness context, which means first and foremost approaching the problem of infrastructure integration with one's sights on meeting the ambitious challenges set forth in this paper. As was noted earlier, the architectures to support the new ebusiness applications require augmentation of the basic multitier architectures we've seen to date. The requirement now is for a comprehensive, network-based, ebusiness infrastructure that incorporates all the functions required to achieve successful services-based integration.

In summary, the specific "calls to action" for IT professionals and the organizations they support include:

1. A thorough assessment of legacy integration needs
2. A careful look at the purposing of ebusiness solutions in order to determine load requirements and, therefore, the technologies required to meet them
3. An informed choice of development and deployment tools and organizational changes and goals to support new paradigms for developing ebusiness applications
4. A choice of enabling technologies based on a determination of application needs of the organization, taking care to assess whether the chosen technologies are truly standards based
5. Making the cultural, organizational, and infrastructural changes to support component-based development and deployment
6. Committing to the changes, technological and otherwise, that will accelerate project life cycles, a requirement to maintain competitive advantage in an ebusiness world
7. Determining which applications will be developed internally versus externally, based on factors such as available internal skills, relative costs and schedules, and the competitive advantages reflected by specific applications.

8. Putting in place the means of measuring how applications, and the ebusiness infrastructure that supports them, perform in terms of customer and business partner satisfaction and the organization's bottom line.

A proactive and intelligent approach to the challenges presented by ebusiness infrastructure integration can yield a substantial increase in the quality of service (QoS) achievable relative to the needs of participants in an ebusiness activity. Efficiency, greater participant satisfaction, and, ultimately, more impressive bottom-line performance are all possible. However, while taking the appropriate actionable steps regarding ebusiness infrastructure can yield a high return, addressing QoS involves more than just putting this infrastructure in place.

The next IDC Brief, entitled "Quality of Service for eBusiness: The Impact of Infrastructure," will provide a comprehensive view of the specific issues that IT management must address in order to deliver QoS consistent with a competitive ebusiness presence. The focus will be on defining what is meant by quality of service within the context of ebusiness, performance issues, creating a satisfactory user experience, methodology-based approaches to applying best practices to the development life cycle, and other topics.