



Infrastructure considerations for service-oriented architecture.

Contents

- 2 Introduction**
- 3 SOA primer**
 - 3 Service-oriented architecture
 - 4 Benefits of SOA
 - 6 Web services and SOA
 - 8 Enterprise service bus (ESB) and SOA
 - 9 The IBM SOA Foundation
- 12 Infrastructure considerations for SOA**
 - 14 SOA infrastructure design and enablement
 - 15 Enterprise integration for Web services and SOA
 - 17 Web services management
 - 19 Infrastructure and Web services security solutions
 - 23 Performance testing and management for SOA
 - 24 Availability management for SOA and Web services
 - 26 Virtualization of systems and service delivery
 - 28 SOA governance
- 31 Summary**
 - 31 SOA infrastructure consulting and middleware services
 - 34 Conclusion

Introduction

If you work in the IT industry, you've probably noticed an increased focus on service-oriented architecture (SOA) in the marketplace. Chances are good that your business may also be doing something with SOA, like a pilot or proof-of-concept. However, SOA is not a new concept but a different approach to an old IT problem: how to design and build systems that are flexible and adaptable to the ever-changing business environment. So what's different about it? Why is it hot right now? Thanks to matured Web services standards, SOA is poised to fulfill promises that other application architecture approaches have only aspired to but never delivered. It also reflects a concerted effort to have a comprehensive, end-to-end architecture that works within and between enterprises in order to respond to increasing business demands.

As businesses evolve to become more agile and responsive, they have found that SOA and Web services are key in enabling this transition. But they have also found that it is not enough to focus on their applications in this transformation and optimization; they also must ensure that their IT infrastructure can support the new architecture.

If recent IT history has taught us anything, it's taught us not to forget about the infrastructure. During their initial foray into the use of the Internet to conduct business, many companies focused on ensuring that their Web sites had the "sizzle" to attract customers but failed to take into account the new dynamics of the environment. As a result, their lack of attention to the infrastructure affected the performance, security and operation of their sites and, in the end, tarnished their brand image and cost them in lost revenue.

Highlights

The adoption of service-oriented architecture (SOA) and Web services frameworks can introduce new challenges for the IT infrastructure that must be addressed.

SOA transforms business operations into a set of repeatable business tasks that are accessible over a network as needed.

As businesses adopt SOA and Web services, these new application frameworks pose new infrastructure challenges regarding governance, management, distributed environments and virtualization that must be addressed. To ensure that the new applications can meet their performance, availability, scalability, security and management requirements, the IT infrastructure needs to be assessed and adapted.

This white paper outlines the infrastructure considerations related to implementing and operating an SOA application. It will interest IT professionals who are planning on implementing SOA applications in their IT environments. It will be of particular interest to those who are responsible for IT infrastructures or data centers, such as directors of IT operations, IT architects responsible for setting technology directions, or IT infrastructure specialists.

Before we outline the infrastructure implications for SOA, let's begin with a brief SOA primer.

SOA primer

Service-oriented architecture

Service-oriented architecture is an IT architectural style that supports the transformation of your business into a set of linked services, or repeatable business tasks, that can be accessed when needed over a network. This may be a local network, the Internet or a geographically and technologically diverse network, combining services in New York, London and Hong Kong as though they were all installed on your local desktop. These services can coalesce to accomplish a specific business task, enabling your business to quickly adapt to changing conditions and requirements.

These services are interrelated through well-defined interfaces and contracts between the functional units of the application. The interface is defined in a neutral manner that should be independent of the hardware platform, the

Highlights

SOA principles are closely related to those of older, object-oriented models but operate on a less granular level.

With the development of XML, WSDL and other languages, SOA has become less complex and more flexible.

With SOA, you can gain the flexibility to enable faster growth, lower costs and better access to information.

operating system and the programming language in which the service is implemented. This allows services, built on a variety of such systems, to interact with each other in a uniform and universal manner.

Service-oriented architectures are not new but are, instead, an alternative model to the more traditional object-oriented models that have emerged in the past decades. Although SOA-based systems do include individual services that can be built with object-oriented designs, the overall design is service oriented. Because it allows for objects within the system, SOA is object based, but it is not, as a whole, object oriented. The difference lies in the interfaces themselves and the granularity at which services are defined. A classic example of a proto-SOA system that has been around for a while is the Common Object Request Broker Architecture (CORBA), which defines concepts similar to an SOA.

However, the SOA of today is different in that it relies on a more recent advance based on Extensible Markup Language (XML). By describing interfaces in an XML-based language called Web Services Definition Language (WSDL), services have moved to a more dynamic and flexible interface system than the older Interface Definition Language (IDL) found in CORBA. Another difference between CORBA and SOA with Web services is the much simpler application programming interface (API) and the fact that SOA services are less granular; therefore, it is easier to understand their function and reuse. Web services are based on well-known, open-standards-based technologies, including Hypertext Transfer Protocol (HTTP) and Simple Object Access Protocol (SOAP). This is in stark contrast to previous technologies, which were complex and more difficult to support.

Benefits of SOA

An SOA lets you build, deploy and integrate services independent of applications and the computing platforms on which they run, making business processes more flexible. This business flexibility can help bring you faster growth, lower total cost of ownership, and better access to timely and accurate

Highlights

Modeling technology allows business experts to play a more extensive role in the creation of business applications.

information. SOA can help deliver additional asset reuse, easier management, and faster development and deployment. Because change is the only thing you can count on in today’s business world, the ability to respond rapidly to customer demands, market opportunities and external threats is more important than ever.

In addition, by raising the level of abstraction in application development from the object to the business service, it is now possible for those closest to the business to model, reuse or recompose existing services to create value for a corporation. With modeling technology that can break down directly into code, there is a great opportunity to let those that best understand business play a greater role than ever before in the creation of business applications.

The benefits of an SOA are also realized by both business and IT as outlined below.

Business benefits	IT benefits
Business flexibility provided by increased granularity of processes enabled through services	Becoming a more responsive IT organization with a more secure and managed integration environment
Ability to more quickly create business processes and composite applications to respond to changes in the marketplace	Decreasing development and deployment cycle times through the use of prebuilt, reusable services building blocks
Improved customer service using services with reduced worry about the underlying IT infrastructure	Reducing complexity and maintenance costs with common services
	Enhancing existing IT systems rather than replacing them

SOA can make it easier to establish connections with business partners, customers and employees.

SOA can make it less expensive for many companies to link their business processes to those of their suppliers, vendors and other business partners. It also helps make it possible for individual customers and employees to buy or interact online with merchants and a host of other services, whatever their IT platform.

Highlights

Web services technology allows applications to communicate regardless of platform and programming language.

Once Web services are designed and built by technical people, business-process architects can aggregate them to solve business problems.

And as if that weren't enough, SOA provides extraordinary flexibility by treating elements of business processes as components to be reused and combined in different ways to address changing needs and priorities. As a result, SOA helps make companies more agile and responsive to change.

Web services and SOA

Web services are technologies that allow applications to communicate with each other in a manner that is platform and programming language independent. A Web service is a software interface that describes a collection of operations that can be accessed over the network through standardized XML messaging. It uses protocols based on XML to describe an operation to execute, or data to exchange, with another Web service. A group of Web services interacting together in this manner defines a particular Web service application in an SOA.

Web services use XML based on open standards that can describe any and all data in a truly platform-independent manner for exchange across systems, thus moving toward loosely coupled applications. Furthermore, Web services can function on a more abstract level that can reevaluate, modify or handle data types dynamically. So on a technical level, Web services can handle data much more easily and can allow software to communicate more freely.

On a higher conceptual level, we can look at Web services as units of work, each handling a specific functional task. One step above this, these tasks can be combined into business-oriented tasks to handle particular business operational processes. This, in turn, allows nontechnical people to think of applications that can handle business issues together in a workflow of Web services applications. Thus, once the Web services are designed and built by technical people, business-process architects can aggregate them to solve business-level problems. To borrow a car engine analogy, a business-process architect can think of putting together a whole car engine with the car

Highlights

SOA is an abstract concept that relies on XML and Web services to manifest itself in software form.

SOA represents the strategy for executing a given process, whereas Web services provide a good approach for defining the tactics.

frame, body, transmission and other systems, rather than looking at the many pieces within each engine. Furthermore, the dynamic platform means that the engine can work together with the transmission or parts from other car manufacturers.

With universally defined interfaces and well-designed tasks, it also becomes easier to reuse these tasks and, thus, the applications they represent. Therefore, the primary issues that Web services address are the issues of data and application integration and those of transforming technical functions into business-oriented computing tasks. These two facets can allow your business to communicate on a process or application level with your partners, while leaving dynamic room to adapt to new situations or work with different partners.

SOA itself is an abstract concept of how software should be put together. It relies on the more concrete ideas and technologies implemented in XML and Web services to exist in software form. In addition, it also requires the support of security, policy management, reliable messaging and accounting systems in order to work effectively. You can improve on it even further with the addition of distributed transactional processing and distributed software state management.

The distinction between SOA services and Web services lies in their respective designs. The SOA concept does not exactly define specifically how services interact, just how services can understand each other and how they can interact. It is the difference between defining a strategy for executing a given process and defining the tactics for actually accomplishing this. Web services, on the other hand, have specific guidelines on how messaging between services needs to interact; that is, it is the tactical implementation of an SOA model most commonly seen in SOAP messages delivered over HTTP. Thus, Web services are essentially just one of many ways in which an SOA can be implemented.

Highlights

An ESB regulates message flow and translation between services—a vital task in a well-managed SOA.

Using an ESB can help optimize service delivery to improve cycle time, reduce costs and increase IT flexibility while protecting existing investments.

Although SOA is not restricted to Web services, we think that it is best accomplished via this method. Other protocols that also directly implement service interfaces with WSDL and communicate with XML messages can also be involved in SOA. CORBA and IBM WebSphere® MQ middleware can now also participate in an SOA, using new features that work with WSDL. If two services need to exchange data, they will still need to use the same messaging protocol, but the data interfaces allow the same exchanges of information.

Enterprise service bus (ESB) and SOA

To establish proper control of all such messaging as well as to apply the needs of security, policy, reliability and accounting, there is a new software construct that enters the picture of an SOA. This is the enterprise service bus (ESB), which is responsible for the proper control, flow and optional translation of all messages between services, using any number of possible messaging protocols. The ESB is not absolutely required, but it is a vital component of properly managing your business processes in an SOA. The ESB itself can be a single engine or even a federated system consisting of several peer and subpeer ESBs, all working together to enable efficient messaging and mediation in the SOA system. Conceptually, it has evolved from the store-and-forward mechanism found in earlier computer science concepts, such as the Message Queue and distributed transactional computing.

An ESB can help optimize the delivery of information and services to help improve cycle time, reduce costs and improve IT flexibility. Based on open standards and industry-leading technology such as middleware from IBM, an ESB pattern provides interoperability between different platforms, programming models and software architectures. It aids in protecting and enhancing existing investments in applications, services and skills. The consistency

Highlights

The ESB is an architectural pattern that offers a comprehensive, flexible and consistent approach to integration that is complementary to SOA.

IBM views SOA as a holistic relationship between business and IT—SOA encompasses tools and methodologies for capturing business design and using that information to improve the business.

of an ESB pattern enables incremental enhancement of your connectivity infrastructure, helping to reduce up-front investments and drive down maintenance costs. As your business needs change and integration opportunities arise, an ESB approach can help make your infrastructure more flexible, consistent and manageable.

In general, the ESB is considered to be an infrastructure component because it does not host or execute business logic. This is in contrast to components such as service requesters, service providers and business service choreography, the role of which is to handle business logic.

The ESB is not a product but is rather an architectural pattern that offers a comprehensive, flexible and consistent approach to integration that is complementary to SOA. The ESB pattern is founded on and unifies message-oriented, event-driven and service-oriented approaches to integration. It is an integral part of a service-oriented architecture.

The IBM SOA Foundation

What should be clear by now is that SOA is not just about technology. IBM views SOA as a holistic relationship between the business and the IT organization. SOA encompasses the tools and methodologies for capturing business design and using that design information to help improve the business. It also encompasses the tools, programming model and techniques for implementing the business design in information systems as well as the middleware infrastructure for hosting that implementation, the management of that implementation to help ensure availability to the business and the efficient use of resources in the execution of that implementation. It encompasses the establishment of who has authority and the processes that are used to control changes in the business design and its implementation in the information system. And ultimately, SOA accelerates the time to value for all these benefits.

Highlights

The IBM SOA Foundation is a comprehensive architecture designed with an emphasis on maintaining a clean separation of concerns.

The IBM SOA Foundation is a comprehensive architecture and set of offerings, technologies and practices that address all of these things about SOA. Figure 1 illustrates the IBM SOA Foundation as a logical architecture model. The model attempts to decompose the functional underpinnings of the application design. Notice the white space between architecture elements in the graphical depiction of the IBM SOA Foundation. This was deliberate – not just artistic license – and was intended to illustrate IBM’s emphasis on maintaining a clean separation of concerns. This separation enables the architecture to support the flexibility, interchangeability, resiliency, security, capacity and availability needed to support an SOA-enabled business.

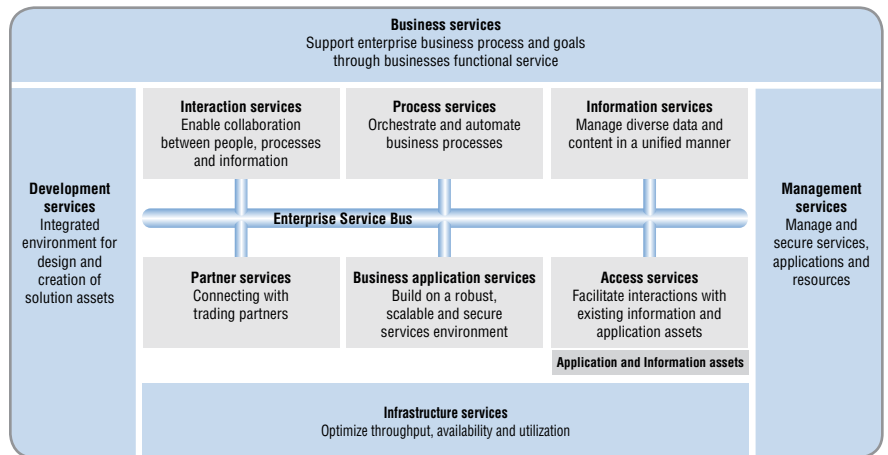


Figure 1
The IBM SOA Foundation

Each element of the IBM SOA Foundation architecture focuses on implementing SOA capabilities.

Each element of the SOA Foundation architecture focuses on implementing SOA capabilities. The gray boxes in the middle of the diagram illustrate those parts that help deploy application software to capture the domain logic specific to the business design. These services include:

- *Interaction services, which are the presentation logic of your business design – components that support the interaction between applications and end users*

Highlights

The SOA Foundation includes interaction, process, business application, information, access and partner services, all of which capture the domain logic specific to the business design.

The architecture includes capabilities to help in modeling of the business design, construction and assembly of the software, deployment of the applications, and management of the operational system and the implemented business design.

- *Process services, which include various forms of compositional logic such as business process flows that automate and coordinate business services*
- *Business application services, which implement core business logic that represents the basic building blocks of your business design—services that are not decomposable within the business model but that can be composed to form higher-level services*
- *Information services, which contain the data logic of the business design and manage your diverse data sources in a unified services approach*
- *Access services, which are dedicated to integrating legacy applications and functions into the service-oriented architecture*
- *Partner services, which capture the semantics of partner interoperability that have a direct representation in the business design.*

The SOA Foundation also includes the ESB, which has already been established as being fundamental to simplifying the task of invoking services—making use of services wherever they are needed, independent of the details of locating those services and transporting service requests across the network to invoke those services wherever they reside within the enterprise.

The blue boxes around the outside of the diagram represent the other parts of the SOA Foundation that exist to assist in the rest of the SOA lifecycle. These parts help in the modeling of the business design, construction and assembly of the software, deployment of the applications, and management of the operational system and the business design that has been implemented. These services include:

- *Business services, which represent the tools and the metadata structures for encoding your business design, including the business policies and objectives*
- *Development services, which encompass the entire suite of architecture tools, development tools, visual composition tools, assembly tools, methodologies, debugging aids, instrumentation tools, asset repositories, discovery agents and publishing mechanisms needed to construct an SOA-based application*

Highlights

- *Management services, which represent the set of management tools used to monitor the service flows, the health of the underlying system, the utilization of resources, the identification of outages and bottlenecks, the attainment of service goals, the enforcement of administrative policies, and recovery from failures*
- *Infrastructure services, which form the core of the information technology environment for hosting SOA applications through the deployment of reliable systems to provide efficient utilization of resources, ensure the integrity of the operational environment, balance workload to meet service level objectives, isolate work to avoid interference, perform maintenance, secure access to confidential business processes and data, and simplify overall administration of the system.*

The IBM SOA Foundation provides a model for IT to deliver the flexibility, responsiveness and efficiency that today's business requires to succeed.

The IBM SOA Foundation provides a model for IT to deliver the flexibility, responsiveness and efficiency that today's business requires to succeed. By viewing the logical architecture model, especially the blue boxes around the outside of the diagram, you can see a set of implications begin to emerge related to both the infrastructure and the operations required to support SOA. But where do you start? Let's take a deeper look at the infrastructure considerations for SOA.

Infrastructure consideration for SOA

Adopting a simpler application framework using an SOA requires the transformation of the infrastructure to support both the SOA and Web services.

As your business adopts a simpler application framework using an SOA, you also need to transform your infrastructure to support both the SOA and Web services. This infrastructure does not change with Web services, but rather it evolves to support Web services. A majority of the infrastructure solutions used today can usually be used to successfully build, secure and manage Web services. The Web services standards, security techniques, application communication and management help define the differences in an infrastructure that is ready to support an SOA. So the real question is, when?

As the applications begin to exploit more mature Web services functions, the infrastructure transformation needs to proceed in parallel to complement the application requirements.

As a result, the techniques used to build infrastructures today need to evolve to build an SOA and manage Web services. This infrastructure transformation occurs along the following eight domains.

Current domains	SOA-specific domains	Description
Infrastructure architecture	SOA infrastructure design and enablement	Efficiently designs and delivers a secure, resilient and variable infrastructure
Integration	Enterprise integration for Web services and SOA	Bridges the gap between the application and infrastructure requirements
Enterprise systems management	Web services management	Manages and monitors Web services to help ensure that they are meeting service requirements
IT security	Infrastructure and Web services security solutions	Builds total solutions using best practices of infrastructure and Web services security
Systems performance	Performance testing and management for SOA	Helps ensure that Web services meet performance requirements for SOA
Systems availability	Availability management for SOA and Web services	Helps ensure that Web services are available, configurable and meeting service levels
System virtualization	Virtualization of systems and service delivery	Delivers computing resources in a flexible, dynamic manner to support Web services
Governance	SOA governance	Puts key IT governance decisions within the context of the lifecycle of business modeling, service modeling and analysis; construction; and business optimization.

If you see a common theme emerging from these considerations, it's that the infrastructure capabilities are evolving from being systems focused to being services focused. This is a fundamental shift in how IT views and delivers functionality, from "I manage a server, I manage a storage area network, I manage a wide area network" to "I manage IT services and components that support this business process." Let's look at each of these infrastructure considerations for SOA in greater detail.

Highlights

One of the domains to be addressed is SOA infrastructure design and enablement.

With SOA, infrastructure architects play a key role and must apply knowledge about the application architecture, functions and even the business processes they support.

SOA infrastructure design and enablement

An infrastructure design for SOA and Web services is still structured around the business, applications and nonfunctional requirements. So in principle, you could tailor an existing infrastructure design to support new Web services requirements. Infrastructure for Web services is about providing the management, security, performance, availability, scalability and operations to support SOA and Web services. Although these considerations are applied today to traditional systems design, they are applied here for SOA with a view toward services as well as systems considerations.

Fortunately, the design methods and reference architectures that infrastructure architects use today are still applicable to building a robust and mature infrastructure design for Web services and SOA.

With SOA, there is an increased importance on the infrastructure architect obtaining the application requirements from the application architect. In the past, infrastructure architects have been able to simply focus on system requirements and avoid having a deep understanding of how applications function. Knowledge about the application architecture, functions and even the business processes they support is key to interpreting infrastructure design and its effect on operations. This is a result of the fact that the application is more virtualized and distributed and may also involve elements provided by business partners. Having this awareness will, in turn, better position the infrastructure architect to design a solution that will meet the nonfunctional requirements of the SOA.

Key questions that an infrastructure architect should be asking include the following:

- *Infrastructure design: How is this application going to be supported within my current infrastructure? How can the design of the infrastructure help ensure that all requirements and expectations are met? Are there new infrastructure components to be added to the landscape, such as an ESB or a service registry?*

Highlights

SOA infrastructure architects should ask a series of probing questions to identify requirements.

- *Application architecture: What are the application requirements, and how will this affect my infrastructure?*
- *Infrastructure and Web services security: How do we secure Web services, and how does this affect my overall infrastructure security goals?*
- *Systems performance: How will Web services affect the performance of my applications? Are we undergoing XML transformation between our legacy and distributed systems?*
- *High availability: What happens if one of my services is unavailable? Where are my applications located, and what are my dependencies between my business partners?*
- *Recoverability: What happens when I need to recover a service? How do I restore state and resynchronize any associated data and metadata to keep data consistency?*
- *Infrastructure and application management: How do I manage my Web services to tell whether they're available and performing? How do I validate my applications and ensure that they're meeting business goals?*

These questions will begin to provide answers for the infrastructure requirements in support of SOA. Each of these question areas is discussed in greater detail in the following sections.

Enterprise integration for Web services and SOA

Enterprise integration represents another domain that should be addressed because it involves creating loosely coupled components that can be deployed dynamically.

One of the goals of implementing an SOA is to provide the capability of integrating a set of loosely coupled, dynamically deployed components implemented as services to meet rapidly changing business demands. But how do you integrate these services that are on multiple, disparate technologies or perhaps even outside the walls of your organization for a business partner?

As mentioned earlier, at the core of SOA is the ESB. This provides the flexibility to integrate and interconnect a wide variety of services to be able to share messages, data and transactions within and across the architecture. Communication services, connectivity services and mediation services are all provided through the ESB.

Highlights

An ESB platform can integrate people, processes and information across the enterprise, allowing access to disparate data sources from a variety of end-user devices.

There are three layers to ESB capabilities: the bus connections layer, the communication layer and the mediation layer.

SOA also contains a set of services that is oriented toward the integration of people, processes, and information to provide the business flexibility to access functions from multiple, different end-user types and devices and to provide access to information and data from multiple, disparate sources, without the need to have direct access and connection to that data.

- *Interaction services provide the capabilities required to deliver IT functions and data to end users, meeting the end user's specific usage preferences, typically implemented through portal technologies.*
- *Process services provide the control services required to manage the flow and interactions of multiple services, typically through the use of business-process choreography technologies.*
- *Information services provide the capabilities required to federate, replicate and transform data sources that may be implemented in a variety of ways.*

This services-oriented approach allows for many of the services to be provided through existing applications; others are provided in newly implemented components, and still others are provided through external connections to third-party systems, such as through business partners. Existing enterprise applications and enterprise data are accessible from the ESB through a set of access services that provide the bridging capabilities between legacy applications, prepackaged applications, enterprise data stores and the ESB.

The ESB capabilities are organized into three layers.

1. *First, we have a bus connections layer that includes a rich set of client APIs, standard protocols and adapters. This layer enables universal connectivity by ensuring that different services may connect to the ESB, whether they are based on the Java™ platform, the Microsoft® .NET platform, or others.*

Highlights

2. *Second, we have a communication layer that provides messaging capability to carry messages between services. It helps provide high-performance, scalable, available and more secure messaging for all involved services, independent of how they connect to the ESB. It provides the flexibility of a range of messaging models as well as a variety of qualities of service.*
3. *Third, we have a mediation layer that provides the semantic glue between disparate services. As mentioned above, we expect services to connect in a number of ways and to be integrated to connect to each other. The mediation layer provides the flexibility to handle data format translation and other transformations necessary to enable communication between services such as XML transformation.*

In addition to these three layers, the ESB offers comprehensive management of services spanning all three layers. Establishing an ESB early on in your adoption of SOA is critical to laying the proper foundation and framework for integration within and among the services.

Web services management

One factor in the ultimate success of SOA lies in the area of services management. You will want to ensure that your enterprise systems management frameworks are extended to include the new SOA applications. Without the ability to effectively manage and monitor these applications, you will not have the necessary insight into your production environments to support SOA. Fortunately, SOA and Web services management are an extension of Enterprise Systems Management (ESM), which most businesses have deployed within their data centers. SOA management hooks into your existing ESM framework to help provide the insight, monitoring and management required to support these new application structures.

Web services management is a key factor in the success of an SOA environment.

Highlights

Designed for development efficiency and speed, Web services technologies are frequently used to build SOAs.

Treating services as manageable resources means applying the entire set of known systems management disciplines to these services – not just to the applications formed by combining these services or to the IT systems whose resources they depend on. Known systems management disciplines include the following:

- *Discovery, monitoring, version and change management*
- *Provisioning*
- *Policy management*
- *Security*
- *Service level agreements*
- *Workload management*

As the environment to be managed is going from monolithic applications to much more granular services, efficient system and service management practice becomes more important.

SOAs are frequently built using Web services, a platform-independent collection of protocols and standards that allows different applications or systems to exchange information. Because Web services are designed for reuse, they help improve development efficiencies and accelerate deployment. To provide effective management, Web services should be incorporated into the end-to-end management domain that supports the composite applications and the SOA infrastructure.

Web services management provides the administration, management, audit and reporting of the deployed Web services in the SOA. This can include:

- *Visualization of Web services through a dashboard/management portal*
- *Analysis of historical Web services data (content and context)*

Highlights

- *Visualization of ESB and messaging infrastructure*
- *Providing views for managing Web services, including their inventory and performance*
- *Providing Web service monitoring views*
- *Start/stop, audit and logging of Web services*
- *Managing Web services using thresholds and quotas*
- *Authentication of Web services, including version checking.*

There are several tools available to help you manage SOA and Web services; it is important to choose the ones that match your operational requirements.

There are various tools in the marketplace to address management of SOA and Web services. The key for most businesses is to understand their operational requirements and ensure that these tools provide the capabilities and functionality needed to meet their service level objectives. Whether you choose to use tools from vendors that specialize in the field of SOA and Web services management or you choose products from end-to-end systems management software like the IBM Tivoli® family of products, most tools usually provide open interfaces to link into your existing enterprise systems management facilities.

IBM Tivoli Composite Application Manager for SOA software can help monitor, manage and control the Web services layer of the IT architecture.

For example, IBM Tivoli Composite Application Manager for SOA software can monitor, manage and control the Web services layer of IT architectures while drilling down to the application or resource layer to identify the source of bottlenecks or failures and to pinpoint services that take the most time or use the most resources. Tivoli Composite Application Manager for SOA software is a core component of IBM SOA Foundation Management Essentials, an integrated and open set of software, best practices, patterns and skills resources to get you started with service-oriented architectures.

Infrastructure and Web services security solutions

One of the most common inhibitors to adoption of SOA and Web services is the lack of secure access to Web services and the lack of secure transmission of data between a client and a Web service.

Highlights

The adoption of SOA and Web services must be accompanied by proper security.

The security environment is still disjointedly hardwired into organizational silos segmented into network security, perimeter security, desktop security, server security and application security. Point solutions solve a partial need, but they don't work in unison. Hence, they can't appreciably lower system risk, improve platform integrity or mitigate the risk of broadening access. Thus, the lack of integrated security management becomes a significant inhibitor to SOA adoption.

SOA adoption introduces new and unforeseen challenges with security integration, identity and security management. These can include:

- *Multiple application platforms, such as WebSphere technology, Microsoft or SAP*
- *Multiple security domains, including internal, external, business unit silos and extranet*
- *Multiple security credentials, such as Kerberos, Security Assertion Markup Language (SAML), Web services security and Resource Access Control Facility (RACF)*
- *Multiple protocols, including SOAP, HTTP, HTTP over the Secure Sockets Layer (HTTPS), Java Message Service (JMS) and Message Queuing (MQ)*
- *Lack of "thread of identity" across the services context.*

Organizations should integrate security and access management silos to mitigate risk.

SOA applications must deal with the challenges of independent security and identity silos. The security solution needs to secure end-user interactions as well as service interactions (application-to-application). Security management needs to provide unified customer views for the composite application. The "thread" of user identity needs to be preserved end to end for auditing and compliance purposes.

Highlights

A service-oriented approach to security is required to provide consistent protection across systems and should be an integral component of service management.

Authentication services help control end-user access.

Identity federation services provide a common platform for validating identities and managing user sessions across companies.

Session management services establish a common view of the user session across channels.

The challenges of security integration across application platforms, business services and infrastructure require that new forms of security and identity services enable SOA applications to leverage “security as services.” The “security services layer” needs to be an integral part of service management. These layers include the following:

- 1. Authentication services deliver identity and authentication services for both passive clients (browser based) as well as active or rich clients, such as desktops, portals and business integration components. Secure access to Web services resources has been largely based on transport level security (to provide confidentiality) using transport security methods such as Secure Sockets Layer/Transport Layer Security (SSL/TLS). For SOA, this must be extended to include the functionality in transport level security by allowing for authentication and authorization of requests based on transport security methods—for example, using mutually authenticated SSL to both build a confidential transport layer and authenticate the requester.*
- 2. Identity federation services is a technology for brokering identities between companies or business units. Federated identity management comprises the set of business agreements, technical agreements and policy agreements that enables companies to lower their overall identity-management costs and improve the user experience. It leverages the concept of a portable identity—the idea that your identity is not bound to a specific credential—to simplify the administration of users in a federated business relationship. Federation simplifies integration because there is a common way to share identities between companies and manage user sessions. Identity federation services within an SOA help ensure that users have simplified access and single sign-on to the composite application environment.*
- 3. Session management services are key to providing consistency. As SOA transactions originate across various channels and protocols, it is important to have a common session management service that enables various SOA components to have a “common view” of the current user session. Such data*

Highlights

Authorization services maintain consistent policies governing communication among platforms.

Auditing services provide visibility into security and change management.

Security-token services translate claims between SOA components that use different types of tokens.

Policy services provide a consistent way of defining, managing and enforcing security policies.

can be used for single sign-on, single sign-off, auditing and reporting and to enable the services to consistently implement policies such as inactivity timeouts and three-strikes-and-out as well as other security policies across various access channels.

- 4. Authorization services help ensure that SOA components can apply consistent authorization policies for Web/HTTP/Java resources, Web services, SOAP (WSDL resources), MQ (queues and queue managers) and even core infrastructure platforms such as UNIX[®] and Linux[®] servers. Authorization services in an SOA help ensure that a common authorization abstraction model enables application platforms such as WebSphere technology, Microsoft .NET, BEA and SAP to apply fine-grained authorization for these resource types.*
- 5. Auditing services provide common auditing services that help ensure that security and change-management activity across the infrastructure and SOA platform can be instrumented, collected, archived and reported for compliance with policies and various regulatory frameworks.*
- 6. Security-token services provide translation services for security claims. SOA applications transcend application platforms that may use different types of security tokens for expressing security claims. These tokens could be binary tokens or XML tokens, and they vary between platforms. This service does the claims translation between various SOA components such as XML firewalls, ESB, Web services platforms (WebSphere technology, Microsoft .NET, SAP NetWeaver) and Business Integrator.*
- 7. Policy services provide a centralized security policy service for centrally defining and managing security policies across HTTP, SOAP (WSDL), MQ and custom resources. This helps ensure that Web services security policies are enforced in a regular and consistent manner across multiple file types, application providers, devices and protocols. This facilitates the secure deployment of Web services, allowing you to deploy your Web services-based applications more quickly and securely in heterogeneous environments.*

Highlights

Performance should never come as an afterthought; it should be engineered into the solution.

Organizations need an integrated solution that will automatically monitor, analyze and resolve end-user response time problems.

Performance testing and management for SOA

By now you should be familiar with the primary benefit of SOA—business agility—but what about its performance? Performance should never come as an afterthought; it should be engineered into the solution. If not properly considered by both the application developers and infrastructure designers, your SOA performance may likely generate inefficiencies.

Because a Web services SOA is mostly XML-based, and XML is text-heavy and verbose, consideration should be given to addressing the challenges of increased metadata. Impact to the network, parsing requirements, security steps (which we've covered) and text translation must be considered when engineering an SOA solution. Beyond application design and optimization, there are various techniques by which the infrastructure designer can deploy specialized tools, such as accelerators, or hardware-based network appliances, such as IBM DataPower® SOA appliances, to help address specific performance bottlenecks—in this case to off-load Web and application servers from the arduous task of XML processing. However, software-based solutions may also provide performance enhancements and the flexibility to customize and upgrade that cannot be provided by hardware solutions.

In addition to performance engineering, performance management functionality must also be considered. Today's business processes often depend on composite applications that span Web servers; Java Platform, Enterprise Edition (Java EE) application servers; integration middleware; and main-frame systems. Although most businesses have traditional monitoring tools to manage individual resources at a high level, many lack an integrated solution that will automatically monitor, analyze and resolve end-user response time problems. As a result, operations and development may take a long time to identify, isolate and fix transaction problems that are negatively affecting customer satisfaction.

Highlights

No matter what management tool you choose, insight into the performance of an SOA application is important for identifying, analyzing and correcting performance issues.

Intelligent availability management tools are essential for identifying and resolving IT problems before they affect your business.

Service level agreements take on new meaning in an SOA, where the service consumer may be within your organization while the service provider is outside.

Tools such as the IBM Tivoli Composite Application Manager for Response Time Tracking product can help you avoid critical Web services performance problems by discovering, isolating and decomposing the Web services calls into the underlying components (Enterprise Java Beans, Servlets, Java Connector Architecture, etc.), allowing for true root-cause analysis of Web services performance and availability failures.

This tool also allows you to follow the path of a user transaction end to end across your business infrastructure. You can drill down each step that the transaction takes as it travels across multiple systems and measure how each component of a transaction contributes to the overall response time. No matter what management tool you choose to use, insight into performance of an SOA application is important in order to be able to identify, analyze and correct performance issues.

Availability management for SOA and Web services

Your IT infrastructure is literally the heart of your business operations, so ensuring the reliability and availability of IT resources is vital for business success. Given the increasing complexity of applications and supporting infrastructures, intelligent performance and availability management tools are essential for proactive identification and resolution of IT problems before they affect your business results. Given the distributed nature of SOA and the dependencies on business partners that may also provide services, any type of single point of failure (SPOF) can disrupt an SOA application's availability.

When implementing an SOA, it becomes even more critical to understand the various aspects of availability and how they are affected by the architecture. Service level agreements take on new meaning in an SOA, where it is possible that the service consumer is inside your organization and the service provider is outside, accessed via a Web service interface. The question is, How do you measure and manage availability in this kind of world?

Highlights

Autonomic systems can dynamically monitor events, analyze behaviors, plan responses and execute policies to help maintain availability.

It is important to understand the availability requirements and characteristics of the service that is being provided or consumed, not just the system. For example, if you need to ensure 24x7 availability of a credit authorization service, then this includes not only the systems or the application involved but also the entire service as defined by the business process. Because an SOA consists of services delivered by loosely coupled applications and systems that can potentially interact within and across organizational boundaries, it is critical to be able to deal with events when they occur in the various infrastructure components that affect the availability of those services.

This is where technologies come into play that monitor and sense faults and process those events through an event-management system. Here policies have been predefined to analyze these events, with a plan or course of action to address the failing component through some change to the environment. Once the appropriate plan is determined, it is automatically executed to repair the fault and maintain service availability. A repair example could be to bring another resource online, isolate and fence off the failing resource, and provide an alternate path to the new resource so that service consumers can still get their service requests processed. This notion of monitoring, analyzing, planning and executing can occur down at the lowest levels of the infrastructure but is part of a hierarchy that includes software, middleware, application and business-process layers.

Consideration needs to be given to put the appropriate infrastructure in place (hardware, software and policies/plans/processes) to be able to sense and capture the failures and provide the plans and resources to dynamically address those failures so that the SOA remains robust and continues to deliver service and so that unscheduled or scheduled maintenance outages are kept to a minimum or eliminated entirely. This is particularly important in an SOA because the notion of using distributed business partners to provide and consume services means that unavailability of those partner services may result in business impact across organizational boundaries.

Highlights

Availability and recovery in an SOA is not just about dealing with service failures but also about configuring and managing the business process that delivers the service.

Virtualization plays a key role in delivering computing resources dynamically when services are requested.

In this case, we need to ensure that appropriate monitoring, measuring and reporting capabilities are in place to know the availability of services in an SOA (be it your own or a business partner's) and a mechanism to understand impacts to availability when changes to the architecture are proposed or injected by any party involved in the service delivery. Availability and recovery in an SOA is not just about dealing with service failures but also about configuring and managing the business process to deliver the service at the appropriate service level (which typically includes performance and availability metrics).

Lastly, availability in the SOA world is about making sure that the systems have the right levels of availability by taking advantage of hardware and software reliability technologies within the servers and storage using spare and fault-tolerant components. This item is no different than availability solutions used today. What is relatively new is providing levels of availability through the use of orchestration, provisioning and virtualization techniques to provide virtual computing resources from among a set of physical machines and using those virtual servers or storage to provide the service when a physical device has been removed for maintenance or because of a failure.

Virtualization of systems and service delivery

In an SOA, virtualization becomes a key enabler for delivering the necessary computing resources in a flexible, dynamic manner when services are being requested and deployed. Virtualization provides the ability to quickly and efficiently use hardware and software resources in order to establish computing capability based on policy and workload characteristics. But what is it, in "real world" terms?

Highlights

Virtualization lets you share resources—such as servers, software and storage—to increase flexibility, improve service levels and reduce costs.

Provisioning and orchestration capabilities can make the infrastructure dynamic enough to support an SOA environment.

Virtualization is about providing a layer of abstraction between the computing resources and those who want to use them. It provides a logical rather than a physical view of data, computing, network and other resources available to applications and users. For your IT teams, virtualization lets you share resources—like servers, software and storage—to increase business flexibility, improve service levels and help reduce costs. Moreover, virtualization is a key enabling technology that can help improve energy efficiency in your data centers by reducing the overall power and cooling requirements. IBM has been a leader in virtualization technologies for more than 35 years. We believe that our approach to virtualization enables IT to be more responsive and flexible to the changing needs of those adopting SOA. For example, in an SOA application, you may wish to execute a specific choreographed business process that makes use of capabilities in the existing infrastructure but requires additional resources because of new workloads in the choreography that were not present when a different path in the business process was taken. A scenario might involve executing a credit authorization when a specific rule is triggered as a result of a threshold being reached in a mortgage application. This may suddenly require more application-serving capacity than was used during normal business processing, and so additional resources may need to be brought to bear. These additional resources could be added using provisioning and orchestration techniques such as:

- *Dynamically adding Web application servers in a WebSphere Extended Deployment software environment*
- *Adding more capacity on the fly to a logical server running in a partition*
- *Adding another logical server in its own partition on a physical computer.*

But it should be noted that virtualization is not only applicable to computing resources but to other IT resources as well, such as storage virtualization, network virtualization and others.

Highlights

The agile nature of an SOA requires organizations to be more deliberate with decision making.

SOA governance involves establishing decision rights and measurement standards across IT to avoid deployments with low business value or negative impact on service delivery.

SOA governance

Governance at its essence is about decision making and communications. These can be large, such as whether to invest in a new program, or mundane, such as providing access to sensitive data or determining whether to include software code in a release. SOAs present a new set of governance challenges as the agility of an SOA requires an organization to be more deliberate in its decision making, perhaps the true core of governance.

SOA enables businesses to be agile and effectively accelerate the creation and deployment of business applications. Hence, it is possible – even likely – that developers will create applications that do not provide business value, or they might deploy applications that negatively affect the overall IT service delivery. To mitigate these risks, IT organizations that adopt SOA need to make decisions more carefully than in the past. Adoption of SOA entails reconsideration of decision rights and measurement across all of the IT disciplines. Organizations adopting SOA need not only to apply governance to the services and composite application lifecycle processes but also to consider the effect of SOA on all of the IT governance subdisciplines, such as portfolio governance or data governance. This resetting of IT governance for IT organizations adopting SOA is called SOA governance.

Governance is critical to an organization's achieving value from its technological investments. The adoption of SOA can act as a catalyst to draw an organization's attention to better governance, IT governance and, of course, service-oriented IT governance practices.

Highlights

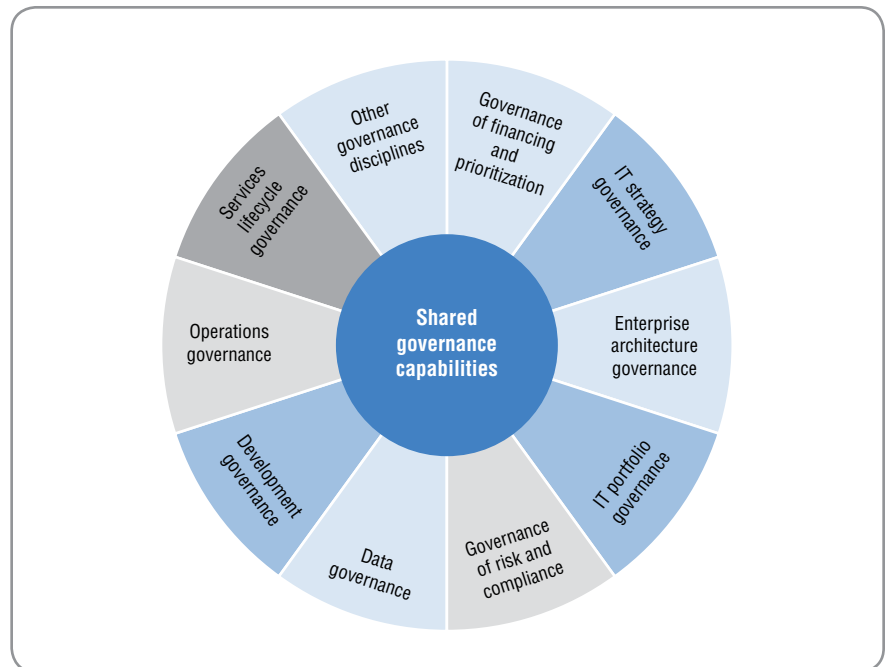


Figure 2
IT governance landscape

A governance landscape encompasses different types of governance and how they are applied across process areas.

To take an operational approach to governance, we need a clear way to reason about the types of governance and how they are related. The kinds of governance and the collection across the areas can be viewed as a “governance landscape.” Figure 2 illustrates an example of governance for an IT organization. This landscape consists of a set of key process areas that are candidates for applying governance.

IT governance addresses decision rights, measures, and control mechanisms to those processes associated with an organization’s business IT function. Generally IT organizations provide value through operational efficiency and internal rate of return on the IT investment; however, hybrids exist. IT governance

Highlights

Organizations can use established capability models to determine how and where to apply governance in the IT organization.

A good governance solution can empower an IT organization to focus on its real work and demonstrate business value.

then includes the decision-making rights associated with IT as well as the measurements, policies, standards and control mechanisms used to guide the way IT decisions are made and carried out within the organization, focusing on the business value that is delivered.

Various IT capability models, such as the IT Infrastructure Library® (ITIL®) and Control Objectives for Information and Related Technology (COBIT), lay out the various processes to be considered in applying governance to IT organizations. Similarly, the IBM Component Business Model™ approach for IT provides a framework for prioritizing which components of the IT organization should be addressed in the governance process.

One of the key areas of IT governance is service-level governance. Service-level governance is tightly focused on the service level agreements – often referred to as key performance indicators (KPIs) – that must be put in place for a process or system to support the business goals and objectives it was designed to support. Key focus in this area is on how to measure the business process to provide the business with critical decision-making information. Without a clear process of how to identify and track the critical indicators, you will have trouble correlating and using whatever information is captured. The usefulness of this information is directly reflective of the process by which it was identified and collected.

To net it out, a good governance solution can empower an IT organization. Well-specified decision rights provide clarity of roles and facilitate communications. Everyone has a clear idea of what communications need to occur to complete a task. Good measures reinforce the right behavior, facilitate the continued execution of the governance solution, and raise confidence that the solution is real and will be followed. In the end, the governance solution should allow IT to focus on its real work and demonstrate business value.

Highlights

Infrastructures must evolve to support Web services and SOA solutions.

IBM can help you transform your IT infrastructure and operations to support SOA.

Summary

With the advent of SOA and Web services, the infrastructure becomes even more important in the overall solution. With a distributed application model such as SOA and Web services, infrastructure capabilities such as security, governance, management, connectivity, performance, integration, high availability and architecture are key cornerstones to a mature SOA enterprise. Infrastructures must evolve to support Web services and SOA solutions. Without any one of these capabilities, the solutions won't withstand production readiness.

The IT transformation is being led by application simplification and componentization. This, in turn, is helping to drive the transformation of infrastructure to support SOA and Web services. Application simplification toward Web services and SOA is a watershed event for IT transformation, including data center infrastructure, operations, hardware and software.

Next-generation IT architectures and service management are focusing on technologies that help simplify the enterprise and support Web services and SOA. Techniques such as virtualization, resource provisioning and orchestration, and increased automation all assist in providing new ways to allocate and manage finite resources to support SOA. Businesses want to do more with less and to simplify their existing environments. How do you begin to transform your IT infrastructure and operations to support SOA? This is where IBM is stepping up to help you address this need.

SOA infrastructure consulting and middleware services

IBM Global Technology Services has a suite of SOA infrastructure consulting and middleware services that have been enhanced to support the SOA infrastructure services lifecycle—from strategy and planning to design and implementation.

Global Technology Services is equipped to cover the core infrastructure services that an SOA deployment affects, such as IBM WebSphere, IBM Tivoli and IBM Information Management software; Microsoft .NET services; integration; testing; security; and SOA governance and management. Typically, SOA

Highlights

IBM offers consulting and middleware services that help you address the eight domains discussed in this paper.

IBM SOA Infrastructure Consulting Services leverages the new architectural approaches and technologies that SOA and Web services bring to infrastructure design.

governance and management is the most immediate infrastructure need. Combining its extensive experience with ITIL and enterprise systems management with its expertise with key software vendors in this space, Global Technology Services is well prepared to help you with your SOA governance and management needs. Global Technology Services is also uniquely positioned to help you more quickly and effectively deploy SOA applications and Web services by addressing infrastructure requirements with proven reference architectures and designs. Tight integration with our WebSphere and Enterprise Services for Microsoft Technologies teams enables smooth and professional deployments, regardless of the infrastructure platform selected for SOA applications.

Global Technology Services' suite of SOA infrastructure consulting and middleware services includes services that take advantage of the new architectural approaches and technologies that SOA and Web services bring to an infrastructure design. The portfolio of services has been conceived to assist you at virtually any stage of your SOA journey.

IBM SOA Infrastructure Consulting Services includes the following:

- *Infrastructure strategy and planning for SOA: This service provides an overall approach for determining the preparedness of your IT infrastructure to support a service-oriented architecture. We evaluate your IT processes, organization, governance and technology using IBM best practices models for an IT operating environment; identify key areas for improvement; and provide recommendations on how to transition your IT infrastructure to support the implementation of SOA.*
- *Infrastructure architecture and design for SOA: This service assists you in developing an architectural framework and infrastructure designs to support a service-oriented architecture. It uses proven IBM reference architectures and leading practices to help accelerate the SOA design process, incorporating state-of-the-art technologies into your existing environment. This service also creates a roadmap that defines projects to transition to an open, integrated, flexible and affordable infrastructure to support SOA. This service*

Highlights

addresses specific design models for applications, infrastructure, data and security features that incorporate new technologies for integration, virtualization and automation.

The middleware services portfolio from IBM includes:

Middleware services from IBM can help support your SOA initiatives.

- *IBM SOA Integration Services – connectivity and reuse is designed to accelerate the adoption of SOA connectivity capabilities, such as WebSphere Enterprise Service Bus, DataPower, Process Server, and Message Broker software. These services focus on the design and implementation of SOA connectivity technologies that are integrated into your heterogeneous IT environment. IBM expertise and repeatable tools enable you to achieve an infrastructure that is deployed rapidly and with a lower risk – and one that uses existing resources to provide users new flexibility and responsiveness.*
- *IBM Application Infrastructure Services provides skilled IBM architects and certified professionals who use leading-edge technologies and methodologies to help install, upgrade or migrate your Web application infrastructure.*
- *IBM Infrastructure Optimization Services – testing center of excellence is designed to help you understand how you can best meet development schedules and reduce costs. Work with IBM consultants to improve, plan for and validate the performance and scalability of applications, systems, storage solutions and networks – both IBM and non-IBM – to support key business and technology initiatives, including SOA.*
- *IBM IT Lifecycle Management and Governance Services helps clients assess, define and implement an integrated set of systems management tools and processes that provide a business-oriented view of how well IT services are performing. Specifically, IBM offers the business of IT dashboard, which*

Highlights

provides the visibility clients need to increase application and infrastructure flexibility to support dynamic business objectives. The dashboard solution enables better IT planning, increases productivity and provides insights that enable CIOs and senior IT managers to respond effectively to changing business conditions. Also, service management implementation services from IBM can help clients improve control over configuration, change and resource management processes.

IBM's service offerings help accelerate your projects.

In addition to Global Technology Services' SOA infrastructure consulting and middleware services, IBM also has a wide range of service offerings available to assist with component business modeling and decomposition as well as SOA application design and deployment. No matter what stage of SOA adoption you are in, IBM most likely has service offerings to help accelerate your projects by leveraging our deep expertise.

Conclusion

Although the move to adopt SOA by many businesses is being spearheaded by their application development groups, the corresponding infrastructure and operations teams must also be involved to ensure that a production-ready SOA infrastructure is in place to support their newly designed applications. This will help ensure that requirements and expectations are met in terms of overall service delivery by both components.

SOA can help your organization be more flexible, responsive and resilient.

If you want to grow your business, service orientation is one of the ways to do it because it allows you to more flexibly link into your business partners, make changes and respond to the marketplace. A service-oriented architecture provides the underpinnings for today's flexible, responsive and resilient

business; if your organization's goal is to become one, then it really can't reach the ultimate potential of that goal without the flexibility that a service-oriented architecture brings to the table. By enhancing your infrastructure to support an SOA, you can change how you typically provide IT services to your applications and to the business as a whole – by moving from a systems to a services focus.

For more information

For additional information, please contact your IBM representative or IBM Business Partner or visit the following Web pages.

For IT strategy and architecture services:

ibm.com/services/itsaconsulting

For middleware services:

ibm.com/services/middleware



© Copyright IBM Corporation 2007

IBM Global Services
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
05-07
All Rights Reserved

IBM, the IBM logo, Component Business Model, DataPower, Tivoli and WebSphere are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

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

Microsoft is a trademark or registered trademark of Microsoft Corporation in the United States, other countries, or both.

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

ITIL is a registered trademark and registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

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

IBM assumes no responsibility regarding the accuracy of the information provided herein and use of such information is at the recipient's own risk. Information herein may be changed or updated without notice. IBM may also make improvements and/or changes in the products and/or the programs described herein at any time without notice.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.