Five best practices for deploying a successful service-oriented architecture

Leveraging lessons learned from the IBM Academy of Technology

Executive Summary

Today's innovative market leaders are in the forefront of implementing advanced business applications—leveraging industry-specific solution frameworks supported by a service-oriented architecture (SOA). In fact, yours may be one of many organizations already using the IBM SOA Foundation[1], which can help improve the integration of your business applications and data while empowering your environment to respond more quickly to demands for new and changing services. As a business leader, you expect efficient transformation, not problems or delayed deployments. Your goal is to see a rapid return on your investment, and when it's time to report to the boardroom, you want the discussion to focus on outstanding results.

[1] Typical SOA Foundation case study: Integration of IBM program products with legacy and newly created applications over the duration of about four (4) months.

Like your organization, IBM strives for high achievement and consistent, measurable results. To this end, the IBM Academy of Technology recently initiated a study of SOA Foundation deployments. With the objective of identifying a set of best practices based on successful outcomes, a team of IBM's top technologists—120 IT architects, developers, IT specialists, and project managers from around the globe—reviewed more than 200 case study submissions, choosing nearly 100 case studies that were selected for evaluation. The team categorized 750 lessons learned and 650 best practices across 10 domain categories identified as the most important focus areas for a successful SOA deployment, including:

- Business process modeling, monitoring, management, dashboards
- Enterprise service bus (ESB) implementation and federation
- Information as a service
- Methods and tools
- Solutions and interoperability
- Meeting nonfunctional requirements
- Rapid delivery of pilot / proof of concept
- Development and operational environment installation, configuration
- Problem determination and resolutions, debugging, testing
- Organizational factors

Focusing on five top priorities

From an executive perspective, five key best practices for a successful deployment consistently emerged from within the ten domain categories, including the need to:

- Develop an architecture with a vision for the future.
- Foresee linkages from IT to your business processes.
- Create an organizational culture and skills to support SOA.
- Build a scalable infrastructure.
- Enable operational visibility through governance and service management.

Taking a broader view of architecture

Looking beyond simple connectiveness and focusing more on architecture is the most common recurring need across the ten categories. The IBM study demonstrates that to achieve a successful deployment, a core architecture leadership team must first be established to ensure consistency of efforts and direct the vision of the architecture. Best practices specific to architecture from this study can be grouped into four distinct themes: reuse, data management, security and the need for a proof of concept.

Reuse

Your objective should be to create a reusable architecture framework to facilitate SOA initiatives across the enterprise by:

- Establishing that the reuse of architectural components takes precedence over individual project requirements
- Assessing new requirements from the perspective of creating reusable components

You shouldn't assume that existing Web services comply with standards or that they can be easily composed and integrated. Service composition is highly complex and often includes transactions, unit-of-work boundaries, error handling, security and propagation of credentials, and aggregation of business logic. As a best practice, you need to design services for reuse at the outset and increase reusability as the business case for the required effort progresses. Most important, remember that reuse requires careful planning, engineering and management.

Message routing and the ESB

Message-level, content-based routing is a pivotal part of any robust SOA infrastructure. However, you don't need to reinvent the wheel. Instead, you can adapt your existing routing protocols, such as TCP/IP, which are known for their robustness and flexibility.

As your SOA implementation matures, an ESB architecture will most likely provide the translation and routing of messages between different services. Your solution should be simple enough to meet requirements, yet be able to adapt to your existing infrastructure. As SOA becomes more widely adopted across your enterprise, the role of the ESB will become more critical, so your solution should be integrated with a common framework and tooling. Avoid letting the applications control the design—keep application-specific business logic and processing out of the ESB. And resist allowing the ESB to become a catch-all for new function that does not fit anywhere else. The simpler the architecture, the more scalable, efficient and flexible your solution will be.

Data management

For SOA, it is especially important to focus on the data aspect of your implementation. Each solution needs to prioritize the establishment of a data management framework that recognizes the components of the framework, its dependencies, roles and responsibilities. Lack of a framework can result in data mismanagement, trusted source issues and threats to data integrity.

Because of their simplicity and ability to support point-to-point data requirements, traditional data transfer approaches are still preferred in some scenarios. However, as you move toward implementing data as a service, effective information metadata management and use of Common Information Model (CIM) is a key critical success factor. Using CIMs can help speed development by enabling you to establish standards and descriptive metadata for information that can be applied to all interfaces, messages, data structures and data transformations to support reuse. A model-driven approach to standardize best practices will also accelerate development and provide for further consistency across interfaces and informational structures. Using this model-driven approach helps reduce the need for transformation, and makes it much easier to design transformations when they are required.

Security

Although an SOA creates an open framework, it can also enhance the security of your data and systems as you implement security as a service. It is important to establish a comprehensive view of security and delegate specific requirements to appropriate points in the architecture. You also need to leverage middleware services to meet key infrastructure requirements for security, logging and monitoring. By isolating these functions from your solutions, you can improve your capabilities in these areas with a consistent point of control and management.

Proof of concept

When there is no prior experience to provide tested information about making a major SOA design decision, it is beneficial to consider doing a proof of concept. You want to avoid a failed or wrong decision, which could result in penalties. Because many SOA transformation projects involve the migration of proven and time-tested systems to new technologies, products and concepts, it is strongly recommended to use the proof-of-concept approach. Make it a best practice to start with a small, more affordable and quick test implementation to help provide the information necessary to make overarching strategic decisions. Consider developing a portion of the functionality that covers a narrow, but horizontal, end-to-end layer of your larger implementation. And, if the business process you are focused on is highly complex, consider conducting small iterations of the final implementation to help verify the applicability of the concepts as you progress.

Creating linkages from IT to your business

The term SOA indicates that your goal is to implement an architecture that transitions IT into the role of a service-oriented provider for your business functions. This goal requires a purposeful effort to link IT to your business processes with a focus on future business process design—that is, you need to envision SOA solutions by examining how the business processes should run, not how they currently run.

A major challenge is that most business analysts do not have the ability to document requirements, processes and services accurately to facilitate the general alignment of business and IT. Business process modeling provides an effective method for tracing life cycles of key business entities to yield simple, flexible business process models with consistent task granularity to more easily gain consensus among different stakeholders, both at the business and IT level. However, your business process model may be error-prone, because many organizations use structured programming tools and programming constructs erroneously. If high-level business process models are erroneous, it is difficult and expensive to refine them into executable business processes. Correcting the process becomes even more expensive if the process model is later changed, with the same errors corrected multiple times when the model changes are propagated to the implementation. Model quality is critical, and any errors should be identified and eliminated from the first iteration.

Component business modeling (CBM) as well as Service-Oriented Modeling and Architecture (SOMA) both support a best practices approach to modeling. CBM helps you analyze your enterprise by first partitioning it into relatively independent, non-overlapping business components to identify opportunities for innovation or improvement. Service-oriented modeling is necessary for the creation of an SOA and for input uses the results of the business componentization analysis as well as business goals and key performance indicators. The output is an SOA which is independent of any specific technology as well as model for how it can best be realized using the appropriate technologies.[2] In the IBM study, almost 25% of the case studies were early adopters of SOMA and indicated that it was key to their success.

It's also important to think outside the boundaries of your enterprise. Be sure to interlock with service partners, since an SOA has high dependencies on other applications. This partner interlock will help ensure that your interfaces are mutually understood at each phase of your deployment to prevent any potential compatibility issues.

Structuring your organization and building skills

The largest number of lessons learned and best practices identified from the IBM study relate to organizational factors including culture, skills, training, teaming, organization structure, decision making, reward systems, collaboration and governance.

As you move toward SOA, you will want to move away from a siloed organizational structure, which tends to limit thinking within the scope of individual projects. Integrating a large number of projects with their own agendas without some central control is nearly impossible and almost always inefficient. A best practice is to establish a design authority to engage technical stakeholders and promote early architecture decision making.

To facilitate the alignment of business and IT, it is important to provide more training—along with easier-to-use technology—to support business analysts who are responsible for documenting the business processes. Do not assume that your business process analysts easily understand service definitions written by developers. Current tools are geared toward a more technical audience, so there is a knowledge gap along with a cultural difference, which means more training is needed to develop IT business analysts.

With SOA projects you will be dealing with short time-frames, challenging objectives, demanding customers, multiple resources in various locations with different skills and ways of working. Especially with a proof of concept, where the project can be easily sidetracked because of other activities, you need laser-sharp focus through a strong project manager who can anticipate issues and accommodate them in a beneficial and proactive way.

[2] *IBM Systems Journal: Service-oriented Architecture, Volume 44*

Scope creep can also be a contributing factor that leads to poor project results. A best practice is to have a control on the scope of exactly what is included in the pilot or proof of concept. Without this, the project can spiral out of control and not achieve the stated objectives. It is important for the project manager to encourage continual feedback, along with frequent review points and demonstrations. The project manager will also need to translate the experience from a proof-of-concept technical jargon into business vocabulary when reporting to stakeholders. This feedback and communication will help drive the project toward successful completion.

SOA enablement requires use of the latest technology features and product versions, so you will need to provide continuous training and mentoring to reduce risk. Time to install products tends to be significantly underestimated. Integrating various software, hardware and services can be difficult and time-consuming, so even though products may be designed to work together, they may not be easy to implement.

Ensuring the scalability of your SOA

Like the case studies evaluated in the IBM study, your SOA implementation will most likely involve hundreds of connections. As much as you can, use known solutions to meet reliability and performance requirements. Then design, test, and retest to confirm that your performance, scalability and interoperability requirements are met. Never deploy a solution without properly addressing and testing these nonfunctional requirements.

A best practice is to utilize standards-based compliance test tools, enforcing compatibility checks as part of the development process. You can also leverage product functionalities such as mock services, which are part of the IBM WebSphere® Business Fabric, to get an early indication if business objectives are being met. Create a baseline for your Web services performance using appropriate instruments and measurements for your Web services so that you have some tangible information to compare and improve upon. Finally, for planning purposes, keep in mind that given the number of variables in a SOA environment, your staff may need to generate a significant number of test cases—and plan for this accordingly.

When lack of requirements, schedule constraints or variable system requirements do not allow for formal capacity planning and server acquisition, you should utilize automation and virtualization to provision a test environment. Finally, always validate your SOA implementation with a technical architecture review from an independent consultant or test lab. This validation will give you the confidence that your architecture and the products selected will operate within the expected performance envelope.

Achieving operational visibility and control

SOA governance is a key critical success factor for achieving sustained benefits of an SOA, helping to ensure the full architectural benefits of reuse. An effective SOA governance platform not only helps business and IT teams better identify which projects contribute most to business goals, but it also empowers employees to work and collaborate more efficiently by clearly defining their roles and responsibilities. For optimal results, SOA governance should be introduced prior to your first major SOA project, during any initial SOA pilot projects. This positions your organization to successfully handle the complexity of a major project rather than treating it as a training mission.

SOA governance should also align with your organization's enterprise architecture program. An enterprise wide and cross-project scope is needed to avoid overlapping and inconsistent governance models embedded within individual projects that may lead to confusion and slow downs.

Publishing capabilities and services is an important part of governance and helps encourage reuse. Establishing a common language within the project, such as a glossary or terminology, makes it possible for executives and team members at any level to identify and leverage existing SOA services, including disparate teams across your organization.

A focus on service management can also improve oversight of your SOA implementation. Problem and incident management systems along with problem determination and troubleshooting utilities should provide a sufficient configurable level of information at all levels of your architecture. At the same time, selecting technology solutions that simplify SOA infrastructure management will help limit complexity. For example, many ESB functions —such as routing, transformation, policy and security—are integrated with a single WebSphere DataPower SOA appliance, simplifying deployment and operations management.

Supporting sustained results

Focusing on the five factors in this paper can help you more efficiently meet your investment objectives as you plan for and deploy an SOA as well as deliver ongoing benefits as you extend SOA across your enterprise solutions. The IBM study reveals success is more than simply designing and implementing new technology. Architecture is more than just connecting your systems and networks—architecture is a vision for how you want IT aligned to your business.

Capitalizing on SOA infrastructure services from IBM

IBM is using the findings of the IBM Academy of Technology study to influence product roadmaps and enhance the IBM SOA Foundation. Best practices from

the case studies and ongoing SOA projects provide continued improvements to IBM Global Services capabilities with increased services standardization to further extend our offering portfolio. Another direct result of the study is our patented architecture decision accelerator for SOA, a tool used by IBM specialists to speed architecture strategy and design. To help you prepare to meet new performance, agility and policy requirements introduced by your SOA-enabled business solutions, our architects and specialists also leverage the IBM SOA infrastructure reference architecture along with IBM Global Services Methods, which incorporate IT service management best practices. Our comprehensive guidance and services can help you:

- Perform a health check of your current IT capabilities to enable SOA.
- Define a strategy and identify initiatives as part of a prioritized roadmap.
- Define an infrastructure architecture that embraces SOA to support your business requirements.
- Develop infrastructure conceptual and specification designs to support the requirements of an SOA application framework.
- Design and implement an ESB to help you decouple your Web services, applications and infrastructure.
- Implement an application infrastructure that supports your SOA applications.

- Plan for a test environment to help uncover performance problems.
- Develop new testing processes and perform complex performance testing.
- Design and integrate a security strategy for your SOA.
- Expand your current service management solutions to support your SOA implementation.
- Leverage virtualization capabilities within your infrastructure.

For more information

For more information about SOA infrastructure services from IBM, part of our IT strategy and architecture services and middleware services portfolios, visit:

ibm.com/SOA/services

IAW03001USEN