

Ten Myths about Service-Oriented Architecture

Summary

The path toward enterprise SOA adoption is often slowed by the myths and misconceptions around it. This article lists ten of the most common myths and attempts to disprove them.

Author: Rohit Kumar Shukla

Company: HCL Technologies

Created on: 17th November 2006

Author Bio



Rohit Kumar Shukla is a SAP NetWeaver Senior consultant working with HCL Technologies.

Table of Contents

Summary.....	1
Author Bio	1
Myth 1: SOA is a purely technical approach.....	2
Myth 2: SOA only means Web services.	2
Myth 3: SOA cannot be built using existing infrastructure.....	3
Myth 4: SOA is a one-time activity.....	3
Myth 5: SOA cannot do business automation	3
Myth 6: SOA doesn't focus on business processes	3
Myth 7: Services will always need to be Complex.....	4
Myth 8: SOA comprises of singular layer automation	4
Myth 9: SOA delivery cannot be planned	4
Myth 10: SOA is not concerned with performance	5
Related Content.....	5
Disclaimer and Liability Notice.....	6

Myth 1: SOA is a purely technical approach.

It is important to understand that service-oriented architecture (SOA), when practiced successfully, is not just technology architecture.

The SOA paradigm is really about modeling the business processes, which are not always supported directly by technology components. Ultimately, services may be implemented by technology components, but the business processes themselves are more important than the services that support them. SOA as a technology is an enabler. The technology doesn't provide direct value. It's not necessarily less expensive to develop services on a line-of-code basis as compared to EJBs or .NET components. Instead, SOA technology should be seen as an enabler of other benefits, such as improved and broader reuse, better responsiveness to changing business processes, and better alignment with business processes.

Myth 2: SOA only means Web services.

A lot of technologists have trouble understanding that SOA doesn't necessarily mean Web services.

Web services can definitely be part of an SOA strategy, but are not required. Service definitions can be based on standard protocols that are not HTTP. It is more important to focus on the needs of the business processes and the services than on the technology used to implement them. Usually, the context of the service will determine how it's implemented.

For example, for services that involve critical business transactions, using a Web service may be detrimental since guaranteeing transactionality across a SOAP/HTTP protocol is difficult. Also, many services may need to operate asynchronously. In these cases, messaging systems based on queues and channels may be a better candidate to implement the service transport. Payloads and interfaces can, of course, still be defined using XML.

Myth 3: SOA cannot be built using existing infrastructure.

Many organizations are surprised to find that they can build SOA using existing infrastructure.

For example, both the .NET and J2EE platforms provide support for developing Web services, for parsing and generating XML, and for communicating with messaging systems such as MSMQ and JMS. What usually is missing in the SOA stack is the process management or automation layer. However, many companies have existing investments in enterprise application integration (EAI) tools. Many of these EAI tools can function as the process automation and management layer, which can access services from existing applications or those built on .NET and J2EE platforms.

Myth 4: SOA is a one-time activity

Service-oriented architecture is not a brand-new solution that comes out of left field and changes everything. Rather, SOA is a natural evolution of both architecture and technology.

Systems architecture has been constantly progressing to become more aligned with businesses. Architects and businesses have long understood the value of aligning technology and business processes, including better use and justification of technology resources and better support of the business.

SOA technology is partly evolved from enterprise architecture theory that has been in place for some time. The SOA tools have evolved from a mixture of Internet technologies, such as HTTP and XML, and integration technologies, such as message buses, translation technologies and connectivity. Enterprise architecture evaluates technology, but more importantly, it looks across the enterprise and across the business and processes. Then it provides a context in which technology decisions can be made.

Myth 5: SOA cannot do business automation

Many organizations and technologists mistakenly focus on the service enablement and delivery within the services architecture.

Unfortunately, that misses the point. The real value of SOA is as a business automation tool. Ultimately, software and systems are about creating business or organizational efficiencies, which can be defined in terms of the processes or activities the organization performs. The focus for SOA should not be on the services, but on the processes and how to improve them.

The services are, of course, necessary to help support the processes. But they're secondary to the goal of making improvements and adding efficiency. Services for the sake of services have limited value.

Myth 6: SOA doesn't focus on business processes

Because SOA is focused on business processes, it's important to understand the data that's relevant to those processes.

For instance, an ordering process has several key data artifacts, such as the order, the customer, the shipping information, the invoice, the payment and the receipt. What's even more important is being able to describe these artifacts in a standard way so each service that participates in the process can understand the data equally.

For organizations with existing information architecture, this may not be a big issue. However, for large organizations with limited or nonexistent information architecture, this issue can be a show-stopper during implementation. Because large organizations have such a variety of data, it is recommended to take an

evolutionary approach to defining the information architecture, as opposed to a big-bang approach. This means that instead of spending four years defining the ultimate data model, it's better to spend a small amount of time during service development to define just the data that's relevant to that service. As each service or process is implemented, the associated information architecture can be evolved to include the necessary data artifacts.

Myth 7: Services will always need to be Complex

In some instances, it's obvious and clear what services are needed. In many cases, the service is reasonably simple — for example, a service to locate a customer might take some criteria that it will use to find the customer and provide a standardized customer record to the service consumer.

However, services can also be compound. This means that "super-services" may provide a standard interface, like the provided by our locate customer service. But our previous example implied that all of the customers were in a single repository, making it easy for the service to search. What if some customers are in a mainframe, some were in SAP, some are in another application, and some are in an Oracle database? And let's assume that we already built service interfaces for locating a customer in each of these systems.

In other words, we have a locate customer service for the mainframe, for SAP, for our application, and for the Oracle database. Our new locate customer service can use all those existing services to locate the customer. Now, because our service is calling other services, it becomes a compound service. Compound services can also be created when an automated process model is itself exposed as a service.

Myth 8: SOA consists of one layer of automation

An often-overlooked characteristic of service architectures is that automation can happen in different layers. Many SOA infrastructures mistakenly look at only one layer for automation; however, automation is usually applied to at least two key areas in an SOA solution.

The first and most obvious area is in the business process layer. When processes are designed, the steps within them are linked together to create the automation. Because these processes are often based on day-to-day business dealings, they tend to involve human interaction. Automation in human interaction processes is one of the key automation layers.

The next most important automation layer is in the system interaction layer. Integration tools have been applied to this layer for the past few years. By automating the tasks that occur between systems, you increase the efficiency of the process overall.

Having different tools for these layers is also important. Different considerations must be made for interactions with a human than for interactions with another application or system.

Myth 9: SOA delivery cannot be planned

There are four main components of a successful SOA delivery plan.

The first component is to define the business processes, what services are required to support them, and what data is relevant across them. These are the business analytics of SOA.

The second component is the SOA architecture and patterns. This is a set of rules that describes how services are defined and implemented, specifies common delivery and usage patterns, and lays out the principles and standards that are adhered to when developing services.

The third component is the SOA infrastructure. This includes the networking, servers, storage, messaging tools, integration tools, process automation tools and so on, which support the development and delivery of services and business processes.

The fourth component is the SOA development program. This program determines the priority for service development and process implementation and guides the projects that result in new services and new processes.

Myth 10: SOA is not concerned with performance

Although SOA does provide a lot of benefits for an organization, including alignment of technology with the business and increased agility, it can have significant performance implications. Because applications are highly segmented in a typical SOA environment, correlating data between applications is often slow. This can be a particular concern in decision support and reporting systems, which historically rely on a small number of data sources.

The key to maximizing performance is in understanding where application and system performance are important to the business. There's little value in building a high-performance system to support a business process that doesn't require it. Once the critical processes are identified, you can focus on enhancement and performance improvements only in the areas where they're necessary.

Related Content

[Enterprise SOA Made Easy \(Really!\)](#)

[Get SOA Expertise from SAP and Partners](#)

[Modeling Concepts in Enterprise Service-Oriented Architecture](#)

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.