

Leveraging Process Configuration within the Context of SOA

David S. Linthicum

[David S. Linthicum Group, LLC](#)

The value of Service Oriented Architecture (SOA) is the concept of agility, or the ability to limit risk through the management of change. Today, many enterprises are fighting to make this a reality, yet few approach SOA using the right methodologies and technology, and most have yet to define the real value of SOA and/or agility.

Truth-be-told, the core notion of SOA is not that complex. In essence, you're looking to identify and define points within your enterprise systems that will be services, information, or processes. From there you define your existing resources, and define the new resources required along the way.

However, the core value of SOA is not to turn everything into services, but the ability to leverage those services to quickly solve business problems. Or, the ability to define and redefine business solutions as the needs of the business evolve or rapidly change. How you approach the creation of this architectural domain is key to the success of your SOA.

In this paper we'll strive to understand the basic approaches to SOA, and how to define and select the process configuration, or "orchestration," technology within the context of your SOA, as well as define the value. We'll look at the basic procedures for understanding your own assets, how to define new assets, and how all of this links into a process configuration layer.

Core to this concept is the use of technology which allows you to configure and reconfigure these processes as the business requires you to do so. In essence, you will drive change through a powerful layer of technology that's able to abstract all related services and information. These layers hold the promise of SOA, and this is where SOA provides the greatest return on investment.

It's about Agility through Configuration

What's core to the value of SOA is the ability to place things that will change over time into a configuration layer, thus making it easier for key business processes to change. For instance, the addition of a new product line causes the way the company defines sales tax to change. Using SOA, we can make that change as a configuration to a process, or process configuration, and not force redevelopment of enterprise systems. Thus, the core notion here is about placing things that will or may change over time into the process configuration layer, and address things that will probably not change as services. This means that the architecture is better able to support change, and thus brings the value of agility to IT.

While SOA was initially sold around the value of reuse, users have come to discover the real value that SOA provides is the ability to change core business processes without requiring waves of redevelopment, testing, and deployment (see Figure 1).

Indeed, agility has proven more valuable than reuse when considering the value of SOA. In a recent study published in *Information Week*, entitled “InformationWeek Analytics: State Of SOA,”¹ it was found that that the value of reuse is only marginal:

“The percentage of overall software reuse within organizations was only marginally higher after initiating SOA, with a 32% reuse rate cited before the SOA project versus 39% after.”¹

Thus, the value proposition around SOA is the ability to promote an agile architecture, or an architecture that is built to change. If we keep this objective in focus, the business value will become apparent, considering the value of accommodating the needs of the business in a much more timely manner.

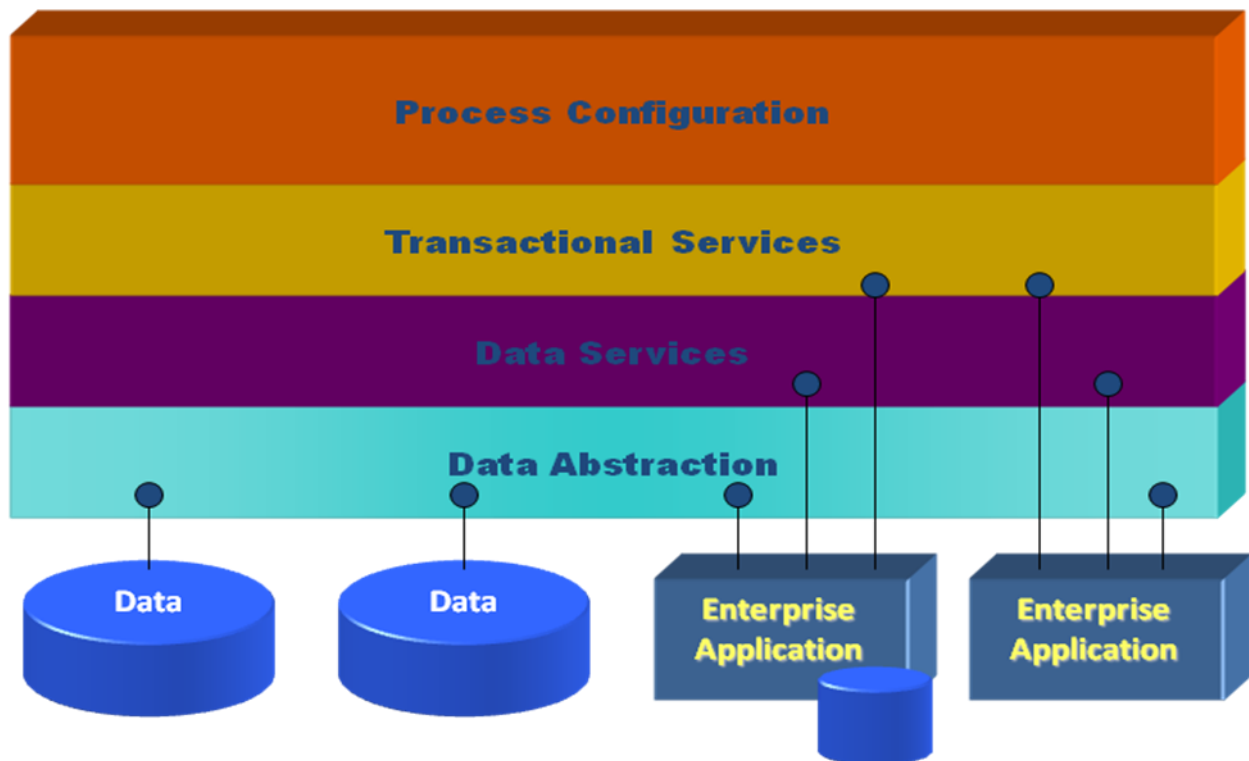


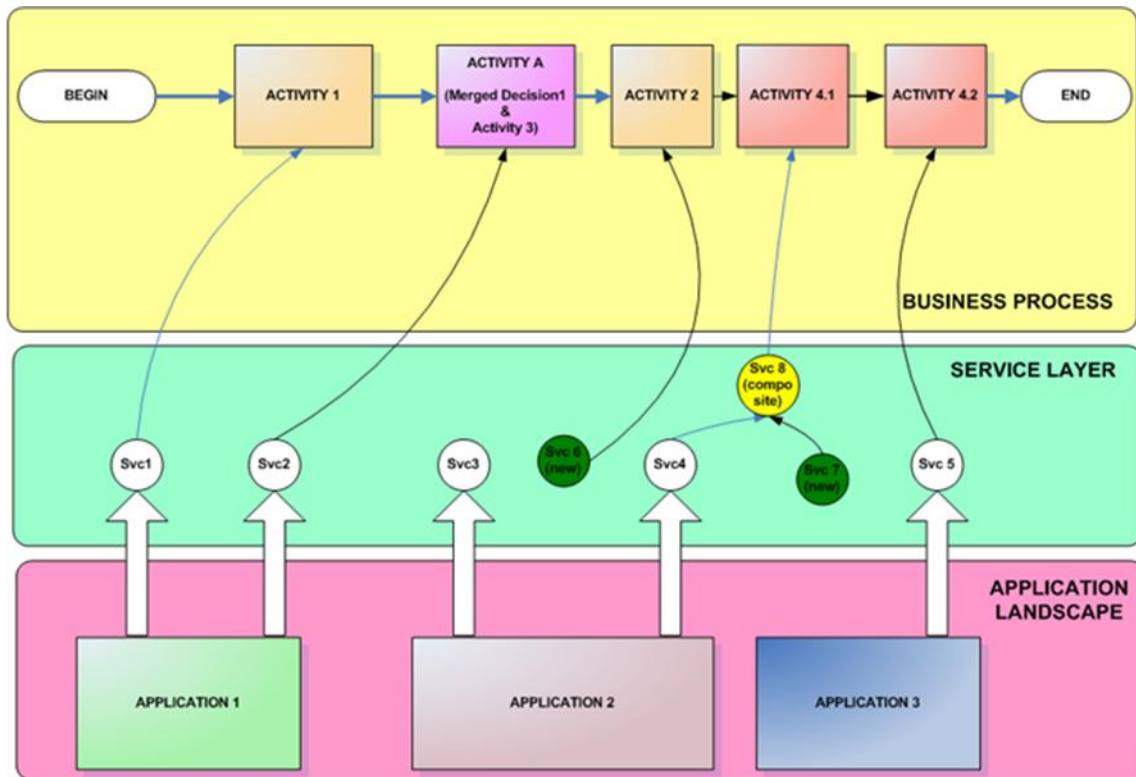
Figure 1: The Process Configuration layer allows you to place volatility, or change, into a single configurable technology, such as a process configuration tool, that supports agility.

¹ “InformationWeek Analytics: State Of SOA,”

<http://www.informationweek.com/news/industry/other/showArticle.jhtml?articleID=214501922>

So, as we define our architecture, we define the data abstraction, data services, and transactional services as largely static. This means that they typically won't change much over time, and services can be added or deleted from the architecture as needed.

In contrast, the process configuration, or orchestration, layer is able to connect with data services and transactional services, exposed from many different business systems within the enterprise, and create processes using process configuration tools, such as [ActiveVOS](#), which is able to abstract these services into a configuration layer where they can be easily and visually bound to processes that provide sequencing and logic control to define the higher-level business processes (see Figure 2).



Source: Wipro

Figure 2: Process configuration tools, such as [ActiveVOS](#), are able to abstract these services into a configuration domain where they can be easily and visually bound to processes providing sequencing and logic control to define the higher-level business processes.

The power of this paradigm is not around the act of defining these processes, but the act of redefining or changing these processes as the needs of the business change over time. In essence, providing a master control mechanism over the core business processes that are able to leverage the power of all connected enterprise systems via services, and thus bring the power of agility to SOA, and to the enterprise.

Considering the value of agility and process configuration as core tenants of SOA, it's a good exercise to determine the value of agility for your business or project. This allows you to better define your own needs, make a business case for the SOA project, and set the appropriate expectations.

So, what's the value of all this? What's the value of agility? Agility is a strategic advantage that is difficult to measure in hard dollars, but not impossible. We first need to determine a few things about the business, including:

- The degree of change over time.
- The ability to adapt to change.
- Relative value of change.

The degree of change over time is really the number of times over a particular period that the business reinvents itself to adapt to a market. Thus, why a paper production company may only have a degree of change of 5 percent over a 5 year period, a high technology company may have an 80 percent change over the same period. Thus, agility has value, but different value based upon what the business is and does.

The ability to adapt to change is a number that states the company's ability to react to the need for change over time. The notion being that the use of a process configuration solution would allow you to make many changes to the core business processes, typically without driving change to the underlying services and data.

Finally, the relative value of change is the amount of money made as a direct result of changing the business. For instance, a retail organization's ability to establish a frequent buyer program to react to changing market expectations, and the resulting increases in revenue from making that change.²

Approaching SOA

When approaching SOA you need to consider that we are not looking to replace systems, just leverage them in different ways to provide IT with the power to change core business processes, at will, when needed, and without a significant impact to cost. In essence, breaking the systems down to their functional primitives, including data and behavior, and building them up again as sets of services that can be further abstracted into process configuration layers that provide the configuration capabilities allowing IT to adjust the IT resources when needed to accommodate the changing needs of the business. While this may seem complex and invasive, with a bit of planning and the right technology, the process of defining and building a SOA is not at all that complex, expensive, or risky.

You need to consider doing the following:

² "SOA Meets ROI," presentation to the Open Group by David S. Linthicum

- Define Application Semantics
- Define Services
- Define Processes

Define application semantics refers to the act of understanding and documenting all information within the problem domain, and what that information means to the business. You must understand the metadata and define the relationships between the data, as well as ownership, security, rules, logic, and policies that may be around the data.

Having a complete semantic understanding of the problem domain means that you have a solid starting point for defining your SOA. From there you can create an abstract representation of the data that may better meet the needs of the business, putting the data in a better logical virtual schema that can be further abstracted into data services (see Figure 3).



Figure 3: Once you define the metadata, you can use database abstraction to create a virtual schema that may better fit the business.

Define services refers to the act of defining the existing and required service behaviors needed for the target architecture. This step is about looking at all system behavior and then determining which behaviors should be exposed as services, or perhaps created. The concept here is to define as many points as needed as services, and do so at a level of granularity that makes sense for the architecture. Keep in mind that services, both data services and transactional services, will become the core interfaces into the architecture, and thus need both governance and security systems in place to manage these services over time.

Define processes refers to the process of defining and listing all business processes that exist within your process configuration domain (see Figure 4). This is important because, now that we know which services and information sources are available, we must define higher-level mechanisms for interaction,

including all high-level, mid-level and low-level processes. In many instance, these processes have yet to become automated or are only partially automated.

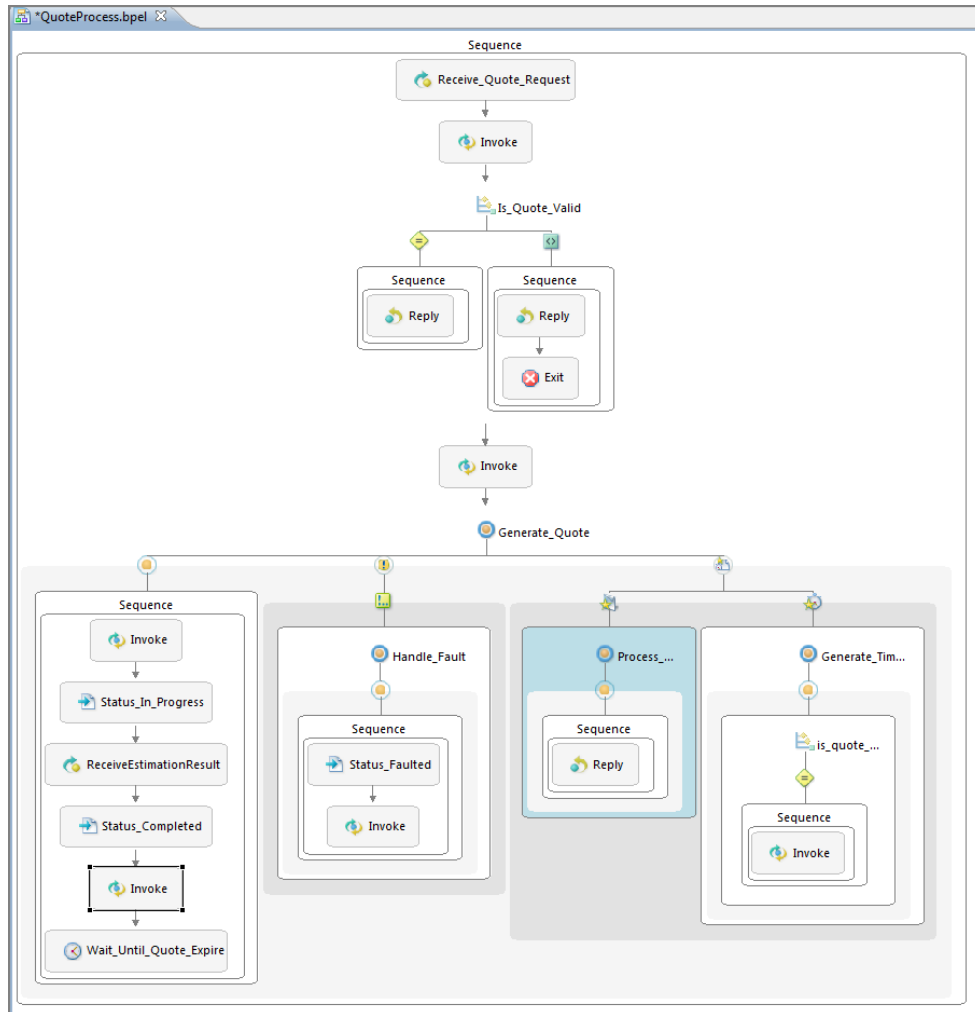


Figure 4: ActiveVOS Designer provides a valuable and cost-effective platform to define processes for a SOA.

The purpose of understanding the processes within the business around the notion of SOA is that the processes are things that will likely change over time, where the services likely won't change. That's a good test to see which behaviors should exist in the domain of a process, and which should exist as services. Keep in mind that processes, typically, can also be exposed as services from the process configuration engine. This provides you with an additional architectural advantage since you can consider processes something that binds services together to solve a business problem, or they can become services unto themselves.

Implementing Process Configuration

Since we're focused on the orchestration (process configuration) layer here, let's drill down on the core issues around implementation. They are:

- Understanding the Use of Standards
- Business Process Management
- Implementing ActiveVOS

We leverage standards with our process configuration layers to reduce the risk and leverage best practices. In essence, this is the ability to provide a base of understanding using well-defined and standard approaches.

[BPEL](#) (Business Process Execution Language) provides a standard language for creation of executable and abstract business processes extends the Web Services interaction model and enables it to support business transactions. The core notion of BPEL is to define an interoperable integration model that facilitates the expansion of automated process integration within the context of SOA. Moreover, BPEL provides a standards-based modeler for BPM, and thus a common language for the definition and deployment of processes. Products such as ActiveVOS are able to make the use of BPEL more manageable by humans, using visual design elements that are both intuitive and pragmatic.

The idea is that the process configuration tool supports a standard, such as BPEL, and the process designer or SOA architect implements that standard within the tool, such as ActiveVOS. In essence, the standard defines the "how," and the process configuration tool, such as ActiveVOS, defines the "what, where, and when."

The best definition of Business Process Management can be found on Wikipedia:

"Business Process Management is a field of management focused on aligning organizations with the wants and needs of clients. It is a holistic management approach that promotes business effectiveness and efficiency while striving for innovation, flexibility and integration with technology. Business process management attempts to continuously improve processes. It could therefore be described as a 'process optimization process'."³

What's relevant in the world of SOA is that the core principles of Business Process Management are leveraged within the context of creating a technical architecture, and within a process configuration layer and enabling technology. In essence, this couples the value of Business Process Management with a core technology architecture, such as SOA. Thus, those who put together the SOA should consider the

³ http://en.wikipedia.org/wiki/Business_Process_Management

core principles and methods of Business Process Management when creating the process configuration layer for their SOA. This will bind an effective approach to process management conceptually, with an effective approach to process management technically.

ActiveVOS provides a process configuration development and execution engine that allows enterprises and developers to automate business processes, collaborate across IT and business boundaries, control the overall state of the business and adapt rapidly and easily to change. ActiveVOS supports the latest versions of BPEL, BPMN and the BPEL4People specifications, which allow human activities to be included in business processes. The use of ActiveVOS provides those who create SOA with the value and ability to change core business processes, as needed, and without significant or any cost. You simply make the changes at the configuration layer, and no additional development, redeployment, or testing is required. With ActiveVOS you can model, simulate, develop, test, deploy and manage sophisticated process orchestrations in a visual designer which creates executable processes.

ActiveVOS allows those who build a SOA to model business processes visually, and then transform them automatically to BPEL for execution. Those designing and deploying processes can leverage the visual aspects of the tool, removing the designer from having to code, or the designer may work from the code as well. Thus, the designer can work at his or her comfort level, making the process of creating or changing processes, quick and easy.

In addition to process configuration and execution, ActiveVOS also provides a dashboard and a universal console with system management capability, including the ability to manage the entire system with dashboards, integrated reports, and a universal console. Also included is enhanced problem determination with “time machine” and “process rewind” capabilities to discover problems in faulted processes.

Call to Action

It is unfair to think that SOA is complex and expensive. If you approach SOA with the right expectations and with the right procedures, the end state should actually result in a significant reduction in operating costs, better efficiencies around core business system processing, and the value realization of agility in your enterprise, or the power to change core business processes as the needs of the business change over time.

However, this does not automatically happen and thus you need to take action to insure that your IT infrastructure and core business systems are optimized for you business requirements. Now is the time to take action and determine the health of your underlying enterprise architecture and determine a path to have IT provide more value. We suggest the following:

1. Understand the cost of the lack of agility when considering your core IT systems.
2. Determine the ROI from leveraging SOA approaches including the use of process configuration technology.

3. Understand all core IT assets within the problem domain including data, services, and processes, and how they are abstracted within a process configuration environment.
4. Define your architecture as an “as is” and a “to be” state.
5. Create the final execution plan and drive the change.
6. Access the value of the change, including the value of process configuration.

There are no magic bullets here. It's really a matter of determining your core business issues and creating an architecture that not only support the business today, but is able to quickly change as the business needs change. While we've been good at creating an instance of enterprise architecture that works for the business during an instance in time, any shift in the business means that the architecture is quickly out of line with the business. We have to make sure that IT can keep up and provide value to the business. The use of SOA and process configuration technology such as ActiveVOS is a huge stride in that direction.

About the Author:

[David S. Linthicum](#) is an internationally known thought leader in the EAI, SOA, enterprise architecture, and Cloud Computing spaces. He is a sought-after consultant, speaker, and writer, and formed David S. Linthicum, LLC (www.davidlinthicum.com), a leading consulting organization focusing on enterprise architecture, SOA, and use of the next-generation Web within the enterprise. He is the former CEO of BRIDGEWERX, CTO of Grand Central Networks, as well as CTO of Mercator Software (now a part of IBM) and SAGA software (now a part of Software AG). In addition, Dave was an associate professor of computer science for eight years, and continues to lecture at major technical colleges and universities, including the University of Virginia and Arizona State University. He keynotes at many leading technology conferences, and has several well-read columns and blogs, as well as a weekly Podcast. Dave has authored 10 books, including the ground-breaking "Enterprise Application Integration." You can reach Dave at david@davidlinthicum.com.