

Mirror, Mirror on the Wall, Whose SOA is the Best of them All?

Mahesh H. Dodani, IBM Software, U.S.A.

1 SERVICE ORIENTED ARCHITECTURE

“Without a good set of pictures, SOA might as well stand for Some Other Architecture. On today’s pictorial evidence, perhaps it does.” – Sean McGrath, http://www.itworld.com/AppDev/1177/nls_ebizsoa050301/

The observation above prompted CBDI (<http://www.cbdiforum.com>) to publish a set of pictures that describe SOA in greater detail (<http://www.cbdiforum.com/>) and communicate the main aspects of SOA (http://www.cbdiforum.com/-report_summary.php3?page=/public/events/workshops/Communicating_SOA.php&area=bronze.) I have attempted to communicate, in previous columns (http://www.jot.fm/-issues/issue_2004_11/column4), IBM’s on demand approach based on SOA. A recent IBM Systems Journal article (<http://www.research.ibm.com/journal/sj/441/crawford.pdf>) provides extensive details on this on demand SOA approach as a comprehensive integration of business process transformation and the enabling technologies of services and policy-based IT management. The above are just two recent examples of the attempt to add more substance and depth to the practice of SOA.

So, what are the common elements of communicating in-depth details of an SOA? There are several key ingredients:

- A reference architecture that provides the details of the SOA. This reference architecture defines the key enabling technologies (services, web services, open standards, etc.), the realization of key SOA principles (composability, loose coupling, reusability, etc.), the details of key architectural layers and components (service provider/consumer architectures, enterprise service bus, component architecture, etc.), and the realization of these components through hardware, software and services.
- Methods to establish a roadmap for transitioning/tranforming an organization from its current environment into the SOA reference architecture. These methods need to articulate how to deconstruct key business processes into executable services and assets that can be realized in the SOA, as well as how to move the organization from its current state to the desired state through a series of transformation steps. A key supporting asset are SOA and service integration

maturity models that facilitate an assessment of the progress of the transformation.

- A governance model for SOA that defines the principles and guidelines, and establishes processes, relationships and organizations to ensure that the SOA is managed effectively and evolves to meet the ever-changing needs of the customer. Key governance processes include conformance, vitality and communication.

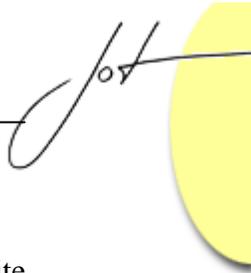
In the following sections, we walk through each of these areas.

2 SOA REFERENCE ARCHITECTURE

There are two types of reference architectures – one built from experience and the other that emerges in addressing a new discipline or area of concern. The proven reference architecture is defined by a set of work products that contains a technical architecture and design template that, when tailored, will solve a class of problems. These reference architectures are derived from successful engagements and are therefore “proven”. Emerging reference architecture, on the other hand, is based on postulated designs in a relatively unknown area with little historical data to reuse and is therefore “emerging”.

Typically, the set of work products describing a reference architecture will include:

- The context under which the SOA will operate. This context includes both the business context that documents the identity of the organization and its interactions with other entities in its environment, and the system context that highlights important characteristics of the system events and data the system receives and generates.
- The requirements for the SOA that describe the users and functional characteristics of the system. The non-functional requirements identify considerations affecting Quality of Service and Constraints for the system.
- The SOA Reference Architecture itself is described using an Architecture Overview Diagram representing governing ideas and candidate building blocks from the perspective of the enterprise, IT, and services. Key architectural decisions leading to choices of components and services are also documented. The description can be augmented with both a Component model that shows how the components interact and work together to provide the required services and an Operational model that focuses on describing the operation of the system derived primarily from the Non-Functional Requirements. Note that the choices of technologies, standards and products (hardware, software, tools, etc.) are documented explicitly in these work products.



Key to detailing the SOA reference architecture is to define the architecture of composite services that align with business processes, as shown in Figure 1 below. The relationship between services and components is that enterprise-scale components (large-grained enterprise or business line components) realize the services and are responsible for providing their functionality and maintaining their quality of service. Choreographing exposed services into composite applications can support business process flows. The integration architecture supports the routing, mediation, and translation of these services, components, and flows using an Enterprise Service Bus (ESB). The deployed services must be monitored and managed for quality of service and adherence to non-functional requirements.

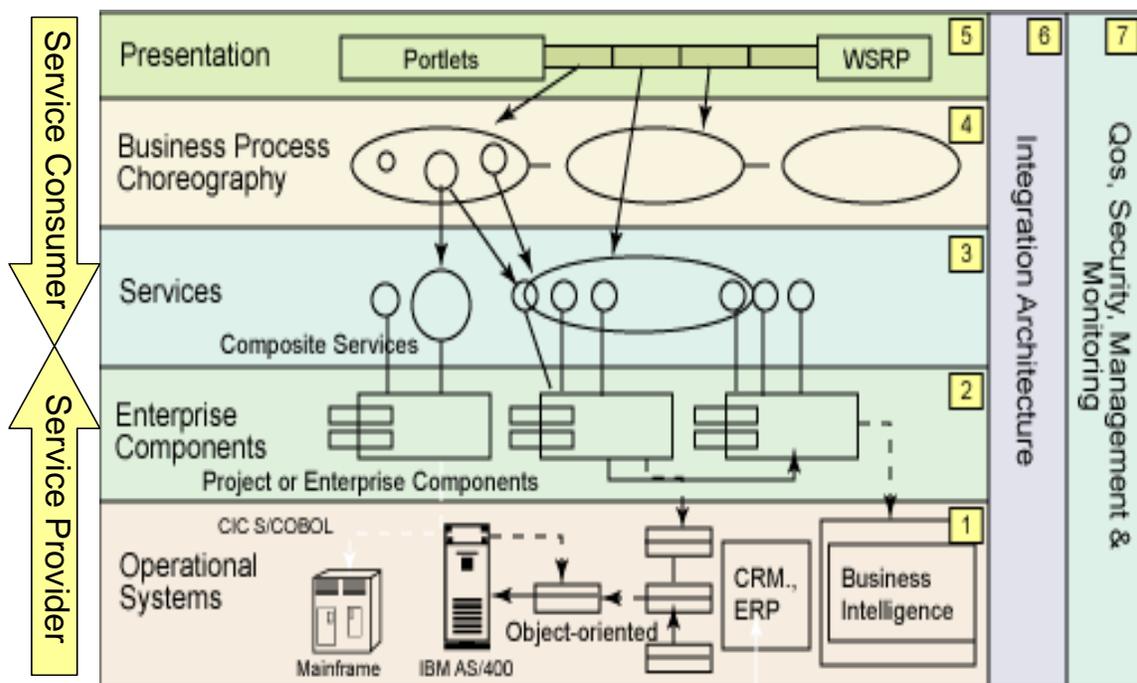


Figure 1: IBM's Layered Service Oriented Architecture

As is evident from the picture, SOA has evolved up from the Service Provider layers focused on building objects and components using object oriented and component oriented approaches, to get integrated with the Service Consumer layers focused on building services and orchestrating services to implement business processes using service oriented approaches.

3 SOA TRANSFORMATION MODELS

The other key ingredient of detailing an SOA is to establish models that help an organization transform their business processes and implement these processes using the SOA. There are several models and approach needed to support this transformation:

- A transformation of the business itself as industry leaders position themselves to adapt and thrive in an environment of continuous, unpredictable change. This requires innovative approaches to model the business through flexible composable processes that can then be realized by the the SOA.
- An approach to implement the high-level business process functionality by choreographing large-grained services. Smaller-grained services that help realize the higher level services are identified by examining the existing legacy functionality and deciding how to externalize the desired functionality via adapters, wrappers and componentization.
- The above approaches follow a business to IT route. For some businesses, it may be more appropriate to follow an IT to business route in transforming themselves to an SOA. This approach builds roadmaps that shows how the infrastructure itself can be transformed from its current state to successive levels of integration maturity – basically moving from silo-ed environments to enterprise integration to integration with known and dynamically defined partners.

IBM uses the Component Business Modeling (CBM) approach to transforming a business (http://www-1.ibm.com/services/us/index.wss/offering_library/igs/a1005119) and break down traditional business silos. CBM can map business strategy to business components, identify key areas of competitive differentiation, and provide insight to opportunities that maximize the cost-effectiveness of non-strategic components. Component business modeling provides a framework for viewing the business as a network of individual components. Once processes and organization are dissected into discrete, understandable, and manageable components, the unique building blocks that make up the company can be identified. Viewing these components autonomously helps decision-makers cut through historical organizational boundaries that have built up through the years, typically along product, channel, customer, geographical, and informational lines. It helps determine when and where resources should be focused. It makes it easier to determine where the real value comes from. It helps define how to source applications. Some may be best provided through existing internal assets or by implementing new systems, while others might be best sourced via commercially available applications packages. Figure 2 shows the main dimensions of the CBM:

- A model that designs the enterprise by aggregating business activities into cohesive and loosely-coupled components that can be shared across the enterprise.
 - Each business component defines a module that contains similar activities supported by appropriate assets, such as people and technology.
-

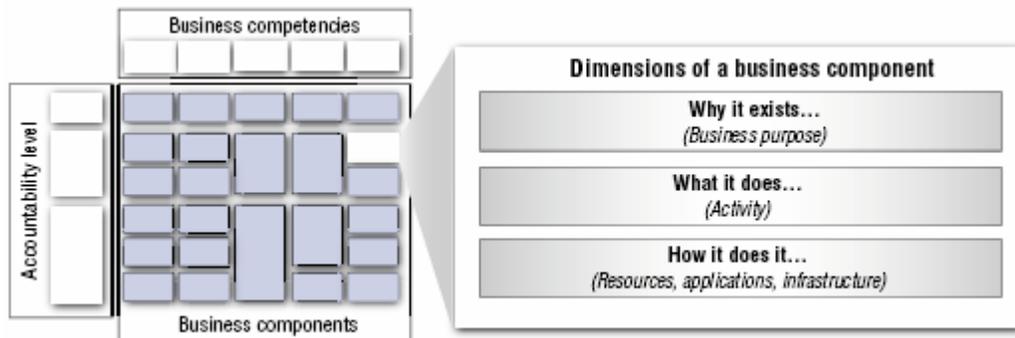
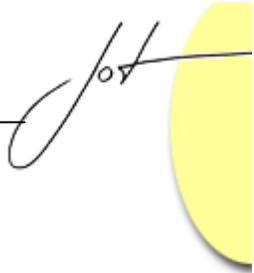


Figure 2: Component Business Modeling

The next step in the transformation is to implement the business components using composable services in the SOA. IBM has defined a Service Oriented Modeling and Architecture (SOMA) methodology that facilitates the identification, specification and realization of services, components and flows through choreography of services (<http://www-128.ibm.com/developerworks/webservices/library/ws-soa-design1/>.) Figure 3 summarizes SOMA.

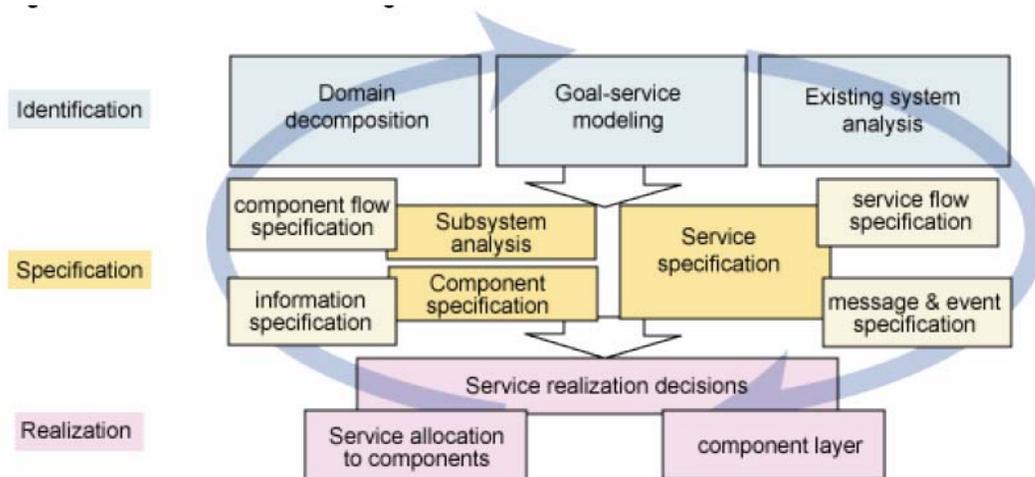
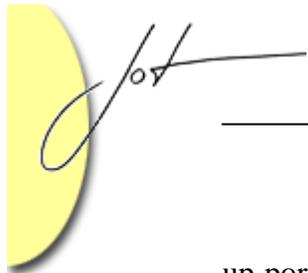


Figure 3: The Service Oriented Modeling and Architecture Process

The service identification process consists of a combination of top-down, bottom-up, and middle-out techniques of domain decomposition, existing asset analysis, and goal-service modeling. In the top-down view, a blueprint of business use cases provides the specification for business services. This top-down process is often referred to as domain decomposition, which consists of the decomposition of the business domain into its functional areas and subsystems, including its flow or process decomposition into processes, sub-processes, and high-level business use cases. These use cases often are very good candidates for business services exposed at the edge of the enterprise, or for those used within the boundaries of the enterprise across lines of business. In the bottom-



up portion of the process or existing system analysis, existing systems are analyzed and selected as viable candidates for providing lower cost solutions to the implementation of underlying service functionality that supports the business process. In this process, you analyze and leverage API's, transactions, and modules from legacy and packaged applications. In some cases, componentization of the legacy systems is needed to re-modularize the existing assets for supporting service functionality.

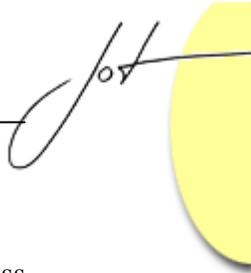
The service classification activity is started when services have been identified. It is important to start service classification into a service hierarchy, reflecting the composite or fractal nature of services: services can and should be composed of finer-grained components and services. Classification helps determine composition and layering, as well as coordinates building of interdependent services based on the hierarchy. Also, it helps alleviate the service proliferation syndrome in which an increasing number of small-grained services get defined, designed, and deployed with very little governance, resulting in major performance, scalability, and management issues. More importantly, service proliferation fails to provide services, which are useful to the business, that allow for the economies of scale to be achieved.

The service realization step recognizes that the software that realizes a given service must be selected or custom built. Other options that are available include integration, transformation, subscription and outsourcing of parts of the functionality using Web services. In this step you make the decision as to which legacy system module will be used to realize a given service and which services will be built from the "ground-up". Other realization decisions for services other than business functionality include: security, management and monitoring of services.

4 SOA GOVERNANCE

Governance defines the rules, processes, metrics, and organizational constructs needed for effective planning, decision-making, steering, and control of establishing an SOA. The governance model defines what to do, how to do it, who should do it, and how it should be measured. Figure 4 illustrates the SOA governance model, which identifies key processes, organizational roles and management responsibilities. The management of the SOA is based on a governance framework whose objectives include:

- The Architecture Management Framework defines the principles, processes, and roles required to manage, use and update the SOA.
- The primary goal of Architecture Management is to derive maximum value from the SOA Reference Architecture asset by promoting its implementation, use and evolution.
- Architecture Management defines how the SOA (and its associated models) should be managed and updated in response to changes in business needs and available technologies.



- The Architecture Management processes are fundamental to enabling the business to make conscious decisions about IT, the acquisition of IT assets, and the design and implementation of new IT solutions to meet business needs

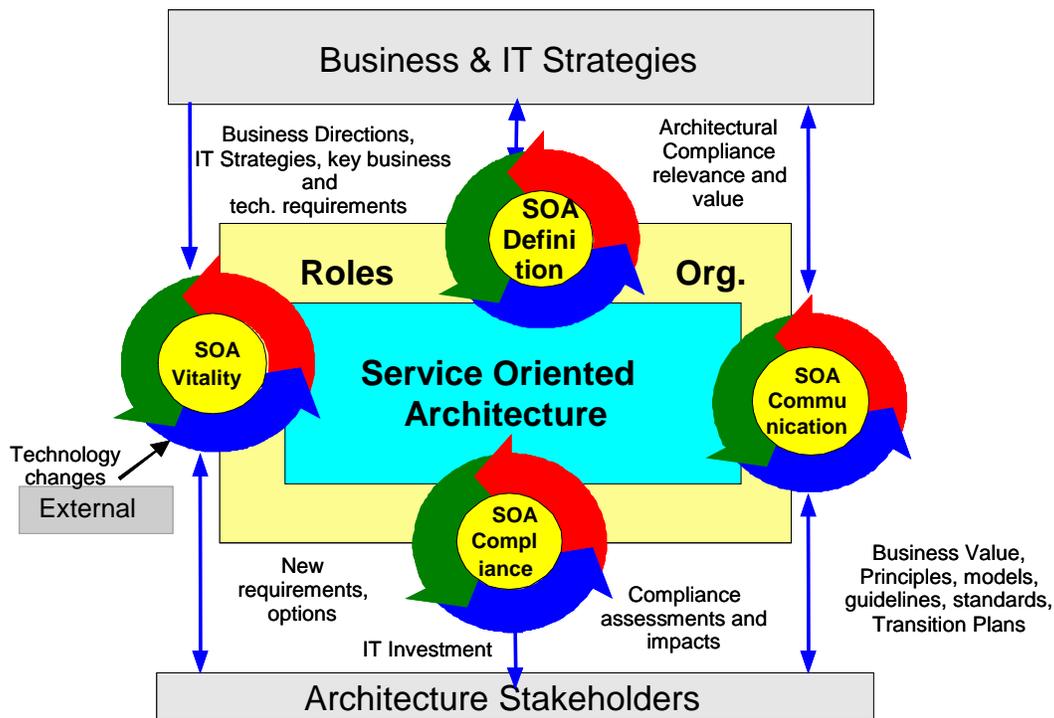


Figure 4: The SOA Governance Model

The governance of the SOA is managed through well-defined closed loop processes involving Vitality, Communications and Compliance:

- Architecture Compliance Process – the process by which the Enterprise Architecture will be used and enforced in the day to day decision making by the Enterprise. To ensure that the value of the Architecture is being effectively captured, both proactive and reactive compliance processes are established.
- Architecture Vitality Process – the process by which new business needs are assessed and the Architecture enhanced to enable them, and by which new technologies are incorporated into the Architecture. Vitality includes the operational (day-to-day) activities required to maintain the currency and accuracy of the existing Architecture base, but does not include the execution of major new Architecture development projects.
- Architecture Communications Process – the process by which the Architecture is communicated, including the appropriate level of education to all of its users.

The IBM best practice on SOA governance is described in more detail in <http://www-128.ibm.com/developerworks/webservices/library/ws-improvesoa/index.html>.

5 CONCLUSION

In conclusion, when looking at the mirror to determine if your SOA approach is “beautiful” make sure you look at it from all angles – the reference architectures that define the models, components, technologies and best practices template for realizing business components or processes; the transformation approach that establishes a roadmap on how to transform your business from its current state to the desired state where these business processes are realized in the SOA; and a governance model that ensures that your transformation conforms to the reference architecture and that the architecture itself is kept vital over the years. These aspects will ensure that your SOA approach is not just “fleetingly beautiful”, but remains so through the years as you transform and evolve your organization to be more flexible and adaptable to ever changing needs.

About the author



Mahesh Dodani is a software architect at IBM. His primary interests are in enabling communities of practitioners to design and build complex on demand business solutions. He can be reached at dodani@us.ibm.com.