



## Implementing the design of service oriented architecture

Dr. Amit Sharma

Professor, Department of Computer Science & Engineering, Vedant College of Engineering & Technology, Bundi, Rajasthan, India

### Abstract

In this research Paper I am focusing on the Today, one of the new solutions to build organizational systems applications, service-based solution is considered. This paper will attempt to identify with the full service approach we offer Integration. In this method, a process model of organization and inventory lists organizations as inputs received during a three-step process for integrating services organizations are recognized organization. Try this method of communication between services is possible and declined to increase communication within the service.

Service Oriented Architecture (SOA) is an architectural style which allows interaction of diverse applications regardless of their platform, implementation languages and locations by utilizing generic and reliable services that can used as application building block. SOA includes methodologies and strategies to follow in order to develop sophisticated applications and information systems.

SOA is different from the traditional architectures as it has its own unique architectural characteristics and regulations, which needs to be analyzed and clarified so as to apply the information that should be included in the architectural model of SOA correctly to service based application development.

This thesis aims to describe the design and implementation of SOA using the web services in detail with considering all the approaches, concepts and methodologies that surrounds architectural model of SOA. Service based application development, service oriented integration approaches, technologies for SOA development, frameworks and other related requirements are discussed in this study in order to have a complete and accurate figure of SOA and be competent in utilizing service orientation concepts in enterprise application development.

**Keywords:** services (common denominator) • service description • advertising and discovery • specification of associated data model • service contracts XML (Extensible Markup Language), UDDI (Universal Description, Discovery, and Integration), Web Service, SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language)

### 1. Introduction

Service Oriented Architecture (SOA) is an architectural style and a combination of methodologies that aims to achieve interoperability of remotely or locally located homogeneous and heterogeneous applications by utilizing reusable service logic. Service orientation shows variation in adopting technology for implementation, rather than focusing on the technology itself, as SOA considers the description of the problem domain before concentrating on the usage of a specific execution environment. Although SOA does not a direct implication of a certain technology and has been applied to software development before the invention of Web services, the capable architectures that realize the SOA vision in a more applicable way are built with Web service technologies. Driven by these competent technologies, the enterprise is practicing open standards for communication over network, messaging and description of service interfaces. SOA with Web services are at the present gaining momentum, as with Web services there is fundamental improvement in SOA based application development.

It is required to follow new approaches and particular methodologies when building service based application structure, rather than tracking the traditional approaches to software development. SOA needs unique development

strategies, which replace the conventional approaches to building software architectures and promise the development of plug-and-play application structures and building modules capable of expressing definite business functionalities and problem domains.

### SOA Components

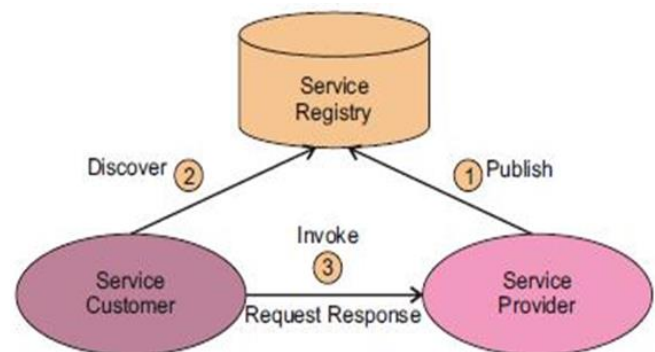


Fig 1: Display the SOA component

SOA provides a strong architectural discipline and focus area centered on service creation, modeling and development,

formation of process information, and service oriented integration approaches and strategies. Services are the building blocks of SOA and new applications can be constructed through consuming these services and orchestrating them within a business process. Services are reusable units for articulating common business and technology functionalities.

To implement a successful SOA in enterprise requires consideration of various concepts and implementation strategies, which formulate the essential characteristics of service oriented enterprise. A complete SOA implementation reflects on not only the deployment of services, but also the possibility of using them to integrate diverse application logs, and building of composite applications.

## 2. SOA problem and solution space

The term service-orientation implies a distinct set of concerns and activities to different audiences. For example, to software engineers it is all about functional requirements, components, integration techniques, messaging, tools, development environments, and middleware. To business people, it is all about implementing business strategies, enabling leaner IT departments, facilitating agile process models, and driving new service delivery processes.

To operational users, it is all about service-level agreements, transparency, flexibility, ubiquitous access to services, and most importantly applications that ease their lives (e-government, e-health, and entertainment).

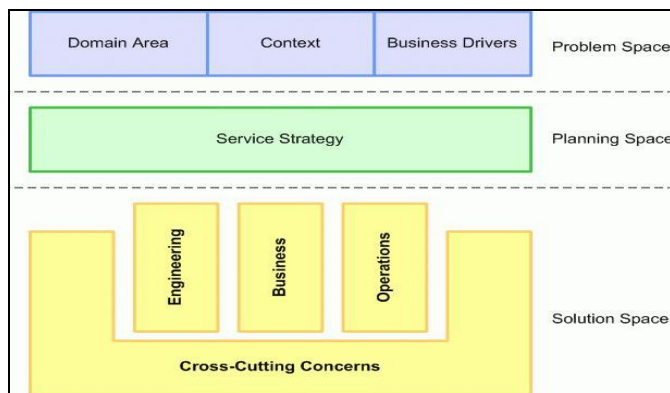


Fig 2: Overview of the SOA problem space and solution space

In a service-orientation adoption setting, an organization should develop a service strategy that takes into account the organization's business drivers, context, and application domain. In order to accomplish its service strategy, the organization has to generate plans to achieve the goals and objectives outlined by the strategy. The execution of these plans requires engineering, business, and operations decisions to be made by the groups identified previously, taking into consideration cross-cutting concerns such as governance, security, risk management, social and legal issues, and training and education. These relationships are shown as problem, planning, and solution spaces in Figure 2.

### Problem Space

The problem space corresponds to the characteristics of the adopting organization as well as the problems that SOA is

expected to address. The problem space shapes and places constraints on the strategy but can also enable the execution of the strategy. The elements of the problem space become the drivers for the strategy.

## 3. Web service technology

### 3.1 Web Services Description Language (WSDL)

The Web Services Description Language (WSDL) forms the basis for the original Web Services specification. The following figure illustrates the use of WSDL. At the left is a service provider. At the right is a service consumer. The steps involved in providing and consuming a service are:

1. A service provider describes its service using WSDL. This definition is published to a repository of services. The repository could use Universal Description, Discovery, and Integration (UDDI). Other forms of directories could also be used.
2. A service consumer issues one or more queries to the repository to locate a service and determine how to communicate with that service.
3. Part of the WSDL provided by the service provider is passed to the service consumer. This tells the service consumer what the requests and responses are for the service provider.
4. The service consumer uses the WSDL to send a request to the service provider.
5. The service provider provides the expected response to the service consumer.

### 3.2 Universal Description, Discovery, and Integration (UDDI)

The repository shown in the above figure could be a UDDI registry. The UDDI registry was intended to eventually serve as a means of "discovering" Web Services described using WSDL. The idea is that the UDDI registry can be searched in various ways to obtain contact information and the Web Services available for various organizations. How much "discovery" was ever used is open to discussion. Nevertheless, even without the discovery portion, the UDDI registry is a way to keep up-to-date on the Web Services your organization currently uses. It can be used at design time and with governance. An alternative to UDDI is the ebXML Registry. More on Universal Description, Discovery, and Integration.

### 3.3 SOAP

All the messages shown in the above figure are sent using SOAP. (SOAP at one time stood for Simple Object Access Protocol. Now, the letters in the acronym have no particular meaning). SOAP essentially provides the envelope for sending the Web Services messages. SOAP generally uses HTTP, but other means of connection may be used. HTTP is the familiar connection we all use for the Internet. In fact, it is the pervasiveness of HTTP connections that will help drive the adoption of Web Services. More on SOAP and Messaging.

## 4. Service oriented architecture

Service Oriented Architecture (SOA) has several core ideas that should be addressed in your organization's SOA journey:

- A set of services that a business wants to provide to their customers, partners, or other areas of an organization

- An architectural style that requires a service provider, mediation, and service requestor with a service description
- A set of architectural principles, patterns and criteria that address characteristics such as modularity, encapsulation, loose coupling, separation of concerns, reuse and composability
- A programming model complete with standards, tools and technologies that supports web services, REST services or other kinds of services
- A middleware solution optimized for service assembly, orchestration, monitoring, and management with the convergence of mobile, social, cloud, and big data analytics, SOA is more important than ever before for offering insight and integrating systems from end to end. By applying Service Oriented Architecture principles, an enterprise can manage and govern business and IT transformation, setting them apart from their competitors. The benefits range from seamless integration, cloud enabled solutions, holistic business insight and agility to externalized APIs. SOA integrates the front office, back office and the Internet of Things. Middleware, best practices and patterns speed the Service Oriented Architecture journey and amplify the value it creates. IBM has over a decade of experience with SOA and a broad portfolio of capabilities, spanning integration, processes, operational control and services. SOA is —simply good design" — resting on a solid foundation of technology and practices that support your organization's journey into the changing world of mobile, social, cloud and big data.

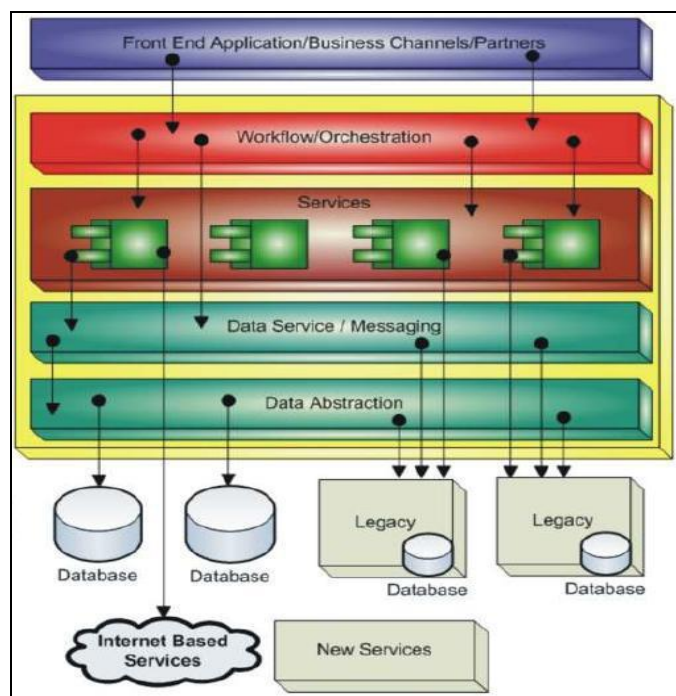


Fig 3: Service oriented architecture view

## 5. Conclusion

In order to fulfill the main objective of identifying new patterns, certain domain areas were identified as most error

prone areas of SOA implementation, hence probable areas of finding new patterns. Service for an application, Discovery of web service through UDDI, and Using plain WSDL for service description have been identified as bad practices often used in SOA projects. The proposed patterns need to be formalized and accepted in industry. Few more bad practices can be identified in the areas of service versioning, data handling and security, since these are the areas where SOA implementation is facing revulsion.

## 6. References

1. Baskerville RL, Cavallari M, Hjort-Madsen K, Pries-Heje J, Sorrentino M, Virili F. The strategic value of SOA: a comparative case study in the banking sector. *International Journal of Information Technology & Management*. 2010; 9:30-53.
2. BEA S. BEA SOA domain model, 2005. Available: [http://www.ebizq.net/white\\_papers/6196.html](http://www.ebizq.net/white_papers/6196.html)
3. Legner C, Heutschi R. SOA Adoption in Practice - Findings from Early SOA, in *Proceedings of the Fifteenth European Conference on Information Systems (ECIS 2007)*, St. Gallen, Switzerland. 2007; pp. 1643-1654.
4. Lam W. Investigating success factors in enterprise application integration: a case-driven analysis. *Eur J Inf Syst*. 2005; 14:175-187.
5. Hirschheim R, Welke R, Schwarz A. Service - Oriented Architecture: Myths, Realities, and a Maturity Model, *MIS Quarterly Executive*. 2010; 9:37-48.
6. Viering G, Legner C, Ahlemann F. The (Lacking) Business Perspective on SOA—Critical Themes in SOA Research, in *Wirtschaftsinformatik Proceedings*. 2009; Paper 5.
7. Schepers TGJ, Iacob ME, Eck PATV. A lifecycle approach to SOA governance, in *Proceedings of the 2008 ACM symposium on Applied computing*, Fortaleza, Ceara, Brazil. 2008, pp. 1055-1061.
8. Niemann M, Eckert J, Repp N, Steinmetz R. Towards a generic governance model for service-oriented architectures, in *Proceedings of the Fourteenth Americas Conference on Information Systems (AMCIS 2008)*, Toronto, ON, Canada, 2008.
9. Ordanini A, Pasini P. Service co-production and value co-creation: The case for a service-oriented architecture (SOA). *European Management Journal*. 2008; 26:289-297.
10. Antikainen J, Pekkola S. Factors influencing the alignment of SOA development with business objectives, in *Proceedings of the 17th European Conference on Information Systems (ECIS 2009)*, Verona, Italy. 2009; pp. 2579-2590.
11. Ross JW. Creating a strategic architecture competency: Learning in stages, *MIS Quarterly Executive*. 2003; 2:31-43.
12. Welke R, Hirschheim R, Schwarz A. Service-Oriented Architecture Maturity, *Computer*. 2011; 44:61-67.
13. Weill P, Subramani M, Broadbent M. Building IT Infrastructure for Strategic Agility, *MIT Sloan Management Review*. 2002; 44:57-65.
14. Varadan R, Channabasavaiah K, Simpson S, Holley K, Allam A. Increasing business flexibility and SOA

- adoption through effective SOA governance, IBM Systems Journal. 2008; 47:473-488.
15. Marks EA. Service-oriented architecture (SOA) governance for the services driven enterprise: Wiley, 2008.
  16. Erl T, Bennett SG, Carlyle B, Gee C, Laird R, Manes AT, *et al.* SOA Governance: Prentice Hall, 2011.
  17. Haines MN, Rothenberger MA. How a service-oriented architecture may change the software development process, Communications of the ACM. 2010; 53:135-140.
  18. Kajko-Mattsson M, Lewis GA, Smith DB. A framework for roles for development, evolution and maintenance of SOA based systems, in International workshop on Systems Development in SOA Environments (SDSOA'07). 2007; pp. 7.
  19. Owens T. An Introduction to Scrum: Understanding and Applying the Software: Speedy Publishing LLC, 2014.
  20. Vlaanderen K, Van Stijn P, Brinkkemper S, Van de Weerd I. Growing into agility: process implementation paths for scrum, in Product-Focused Software Process Improvement, ed: Springer. 2012; pp. 116-130.