



Service Oriented Architecture

Provide Flexibility to Business Operations

Abstract

From several years, organizations are facing issues in aligning business applications (in-house software applications or 3rd party software applications) together as per the organizations goals and business requirements. Organizations have some work around for this common problem such as, implementing adaptor (developing customizing API's) for the third party applications or developing its own application from scratch instead of using third party application. However, the solutions are not permanent and provide only minor relief to the organizations. Moreover, they are time consuming, expensive and require high maintenance as well.

Service Oriented Architecture (SOA) is most comprehensive and most recognized solution in today's IT world. SOA is a framework that provides common and reusable business functionalities in a standardized way as a set of services. SOA provides the flexibility to the organizations to coordinate these services as per their business workflow. A service is a reusable unit, which exposes business functions in standardized way and consumed across multiple projects. Since each service is a reusable component, it reduces development cost and offers consistency across applications.

"Consuming a service is always cheaper and quicker than developing it ourselves."

Proteans has helped many ISV's, Software Development Organizations and Software Service Provider firms implement SOA strategy. This white paper highlights how SOA methodology can help organizations in aligning their business applications as per their business operations and how it copes and quickly responds to the changes in business requirements.

Introduction

SOA is an architectural methodology, which provides solution to the integration issues by providing loose coupling among interacting software agents.

SOA is an architecture comprised of services, where a service is a reusable unit, which encapsulates key business logic, and exposes its functionalities to the consumer in a standardized way so that it can be used across multiple projects.

SOA achieves loose coupling by employing two architectural constraints

1. The functionalities should be exposed as set of simple and standardized Interfaces. These interfaces should be available globally to all consumer and other service providers, irrespective of their platform
2. Contract schema delivered to the interface should be descriptive and extensible. By this generic approach, new functionalities can be incorporated in new version of services without breaking existing services

Principles of SOA

- ❖ **Technology Neutrality:** Exposing functionalities to the consumers irrespective of their platform using the standard protocols such as XML and SOAP
- ❖ **Standardization:** Defining service interfaces and contracts using standard protocols such as XML, SOAP and WSDL
- ❖ **Consumable:** Exposing functionalities in simplified and standardized way
- ❖ **Reusable:** Exposing functionalities as set of services
- ❖ **Loose Coupling:** Performing single operation, independent of other operations, with a well defined interface
- ❖ **Meeting Business Needs and Easy to Manage:** Exposing business functionalities at a granularity, this can be easily recognized by the consumer. Services can be easily orchestrated to support business workflow
- ❖ **Adaptability:** Defining descriptive and extensible contract schema hence, new functionalities can be easily incorporated in the newer version of services without breaking existing service
- ❖ **Abstraction:** Abstracting service from implementation



Figure 1: SOA Principles

Advantages of SOA

Service Oriented Architecture offers flexibility and agility to adapt and align new business process changes in the existing business applications. On the other hand, traditional architecture methodologies do not provide enough room to the business applications to respond quickly to the changes in business environment.

SOA offers a great deal to the Business Organizations and Software Service Provider firms, which they can capitalize while developing the business application as well at real time to accommodate business changes. These are:

- ❖ **Responsive to Business Environment Changes:** SOA is more agile in responding to the changes in business requirements. It provides flexibility to quickly rearrange the system workflow by selecting and orchestrating business functions provided by set of services
- ❖ **Consistent:** SOA brings consistency across the enterprise applications by tendering reusable units called Services
- ❖ **Adaptive to New Technologies:** Always ready to adapt latest and best solution. SOA provides high degree of flexibility in improving services performance by replacing older technologies with latest and most effective technologies without affecting consumers systems by abstracting service interface from its implementation
- ❖ **Cutting Maintenance and Integration Cost:** SOA is a cost effective solution. SOA encourages organizations to develop business components as a set of reusable units and further, reduces the integration and maintenance cost by adopting standard protocols such as SOAP, XML, WSDL etc, which require less integration and maintenance effort
- ❖ **Provision for Security:** SOA provides provision for secure and reliable communication by supporting authentication and authorization at every level which, organizations can extend to ensure that their business transactions and information sharing are secured

Dimensions to Measure SOA Benefits

Organizations need to measure SOA benefits across key dimensions like Business, Technology, and Money, before investing in it.

Benefit Measurement Matrix

Business	<ul style="list-style-type: none">➤ Deliver within the timeframe➤ Automate business process➤ Provision to support new business requirements in future➤ Agile in accommodating changes in the business processes➤ Reduce business risk➤ Re-use of business processes➤ Improve Business Efficiency
Technology	<ul style="list-style-type: none">➤ Deliver high quality solution.➤ Provide loose coupling between integrated systems.➤ Provide freedom to choose best technologies (technology independent solution.).➤ Re-use of components.➤ Provide high degree of scalability.➤ Support cross platform functionalities by using open standards (platform independent solution.)
Money	<ul style="list-style-type: none">➤ Cut down maintenance cost➤ Reduce learning, development and testing efforts➤ Ensure optimum utilization and management of underlying IT resources➤ Re-use of resources➤ Reduce integration cost

SOA Development Risks

As no solution is hundred percent foolproof, SOA is not an exception; it has its own risks. However, these risks can be avoided and mitigated if proper precautions and measurements are taken at right time.

- ❖ **Service Management:** As number of services grow, it becomes difficult to manage and monitor services if no steps have been taken care. Reuse of service will remove redundancy. Using registry services can be categorized and searched by name easily for the reuse, and duplicate services can be found and removed
- ❖ **Performance:** Performance could be an issue if a service is written badly and performance measurements have not been taken care while developing the service functionalities
- ❖ **Communication Gap between Development and Operations Team:** Communication gap between development and operations team may lead to undesirable results. Try to have good understanding and coordination between these two teams to avoid risk
- ❖ **Insufficient Security Measures:** Identify the components where security needs to be applied. Insufficient measures will make system more vulnerable. Integrating Business-to-Business application must authenticate partners and data
- ❖ **Service Reliability:** It is important to authenticate and authorize a service, which is not well known before consuming it
- ❖ **Secure and Reliable Communication:** Sharing the service over the public network should have mechanism in place to provide secure and reliable communication, which may need to carry sensitive information from one business application to other. Risk can be mitigated using secure channel and data encryption, while sending the information

SOA Kits from Leading IT Vendors

Leading IT vendors like Microsoft and IBM have already recognized the potential of SOA. In fact, they are helping organizations of all size in aligning business process to make their business more effective and efficient by offering SOA designing principles, best practices, applications, tools and technologies.

Microsoft's Vision

Since 1999, Microsoft has been engaged in helping organizations with their SOA efforts by first announcing its Web Services model and followed-up with .Net Framework release. Microsoft's SOA approach has helped organizations of all sizes in optimizing their business processes and providing greater business agility through use of Microsoft's SOA design principles, best practices, tools, and technologies.

Microsoft's "Middle Out" Model (An Incremental Approach)

Microsoft advocates what it calls a "middle out" approach. In this approach, SOA efforts are driven by strategic vision and business needs, and are met through incremental, iterative SOA projects that are designed to deliver on business goals, one business need at a time.

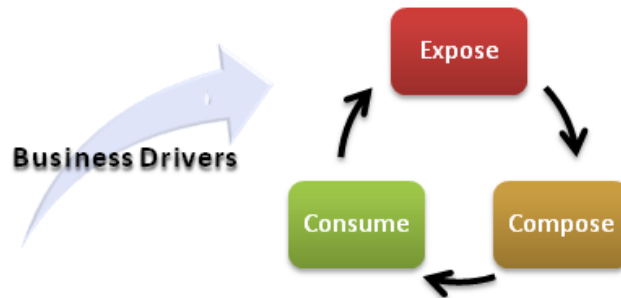


Figure 1: Middle-Out Model

❖ Expose

The expose phase of the SOA approach focuses on which services to create from the underlying applications, data, and IT resources. This phase is also concerned about implementation of these services

❖ Compose

Once services are created, compose phase focuses on orchestration of these services to form more complex services, applications or cross-functional business processes

❖ Consume

Once a new application or business process has been created, that functionality is made available for consumption either other IT systems or by end-users. Users can consume the composed service through a number of avenues, including web portals, rich clients, Office business applications, and mobile devices

Microsoft SOA Platform

Microsoft provides a rich platform for building and managing connected systems. In that effort, Microsoft offers Web Service development tools (like WCF and WF in .Net Framework 3.0/3.5) as well as the server products (like BizTalk Server and Microsoft Office SharePoint Server) for orchestrating and managing business processes.

- ❖ **BizTalk Server:** Its core architecture is based on XML and the .NET Framework. BizTalk Server fully supports all the open standards upon which Web services are built. A BizTalk solution can consume existing Web services and expose business processes (BizTalk orchestrations) as Web services. BizTalk acts as the management layer that orchestrates Web services, controlling the flow and interaction between them and aggregating individual services into a larger composite solution
- ❖ **Microsoft Office SharePoint Server:** Designed to empower users to optimize the way people, content, and processes interact within and across organizations, Office SharePoint Server enables users to take advantage of workflows to automate and gain more visibility into common business activities like document review and approval, issue tracking, and signature collection. Integration with familiar client applications, e-mail, and Web browsers simplifies the user experience

Microsoft SOA solutions help organizations of all sizes integrate and access existing IT resources, assemble them into larger business processes, and make the outputs available to users in order to run their organization more effectively.

3-Tier Architecture versus Service Oriented Architecture

3-Tier architecture contains three layers, Data Access, Business Logic, and Presentation layer where each layer is logically separated from other. Organizations that are building 3-tier applications for years and using these applications across the organization are now facing many issues such as, each application looks and behaves differently and offers no consistency across the organization. In fact, they are difficult to integrate to form a composite application.

3-Tier architecture just provides logical separation between data access, business logic and presentation layer. 3-tier approach is limited to the specific application. It does not provide solution at the enterprise level as its approach is limited to the single application and it does not think beyond the application boundary.

SOA addresses these shortcomings and provides solution at enterprise level. SOA does not look at IT infrastructure; it mainly depends on the services. SOA identifies the different business components and develops each component as discrete and reusable coarse-grained services. SOA uses standard protocols to define service interface, which makes them easy to recognize and consume. These coarse-grained services can be used across multiple projects. SOA orchestrates these services to form the applications or composite applications. SOA provides consistency across the organization by offering re-usability.

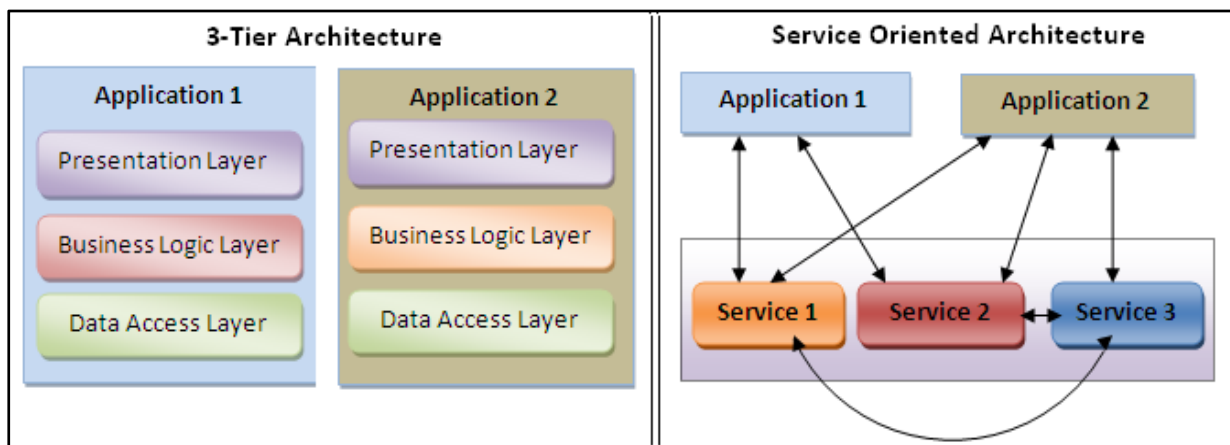


Figure 2

A good example for comparing 3-Tier and Service Oriented approach would be an organization having three different processes i.e. Payroll, Employee and Customer management processes where each process requires authentication and authorization mechanism.

In 3-tier approach, each application will implement its own authentication and authorization component as shown in figure. Developing same component three times will require extra time, effort, and money causing extra burden to the organization.

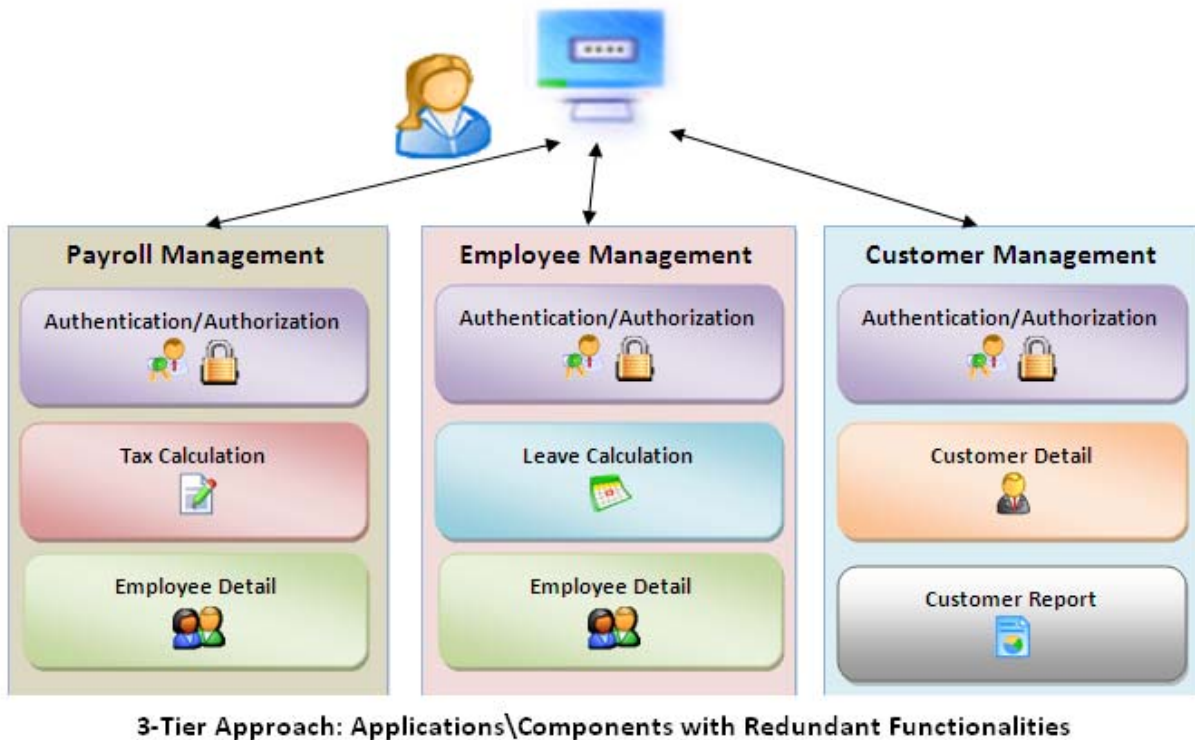


Figure 3

Figure 4 shows a logical view of an organization, which is using three applications developed using 3-tier architecture with duplicate components like 'Authentication\Authorization' and 'Employee Detail'.

In SOA approach, common components like 'Authentication\Authorization' and 'Employee Detail' will be developed only once and functionalities will be exposed as shared services, which can be used in multiple projects across the organization.

Comparing 3-tier and SOA approach as depicted in figure: 4 and figure: 5, if you go by 3-tier approach, the number of components that needs to be developed would be nine, whereas using SOA approach, the count will be decreased to six, since SOA is reducing redundancy by exposing functionalities as reusable units.

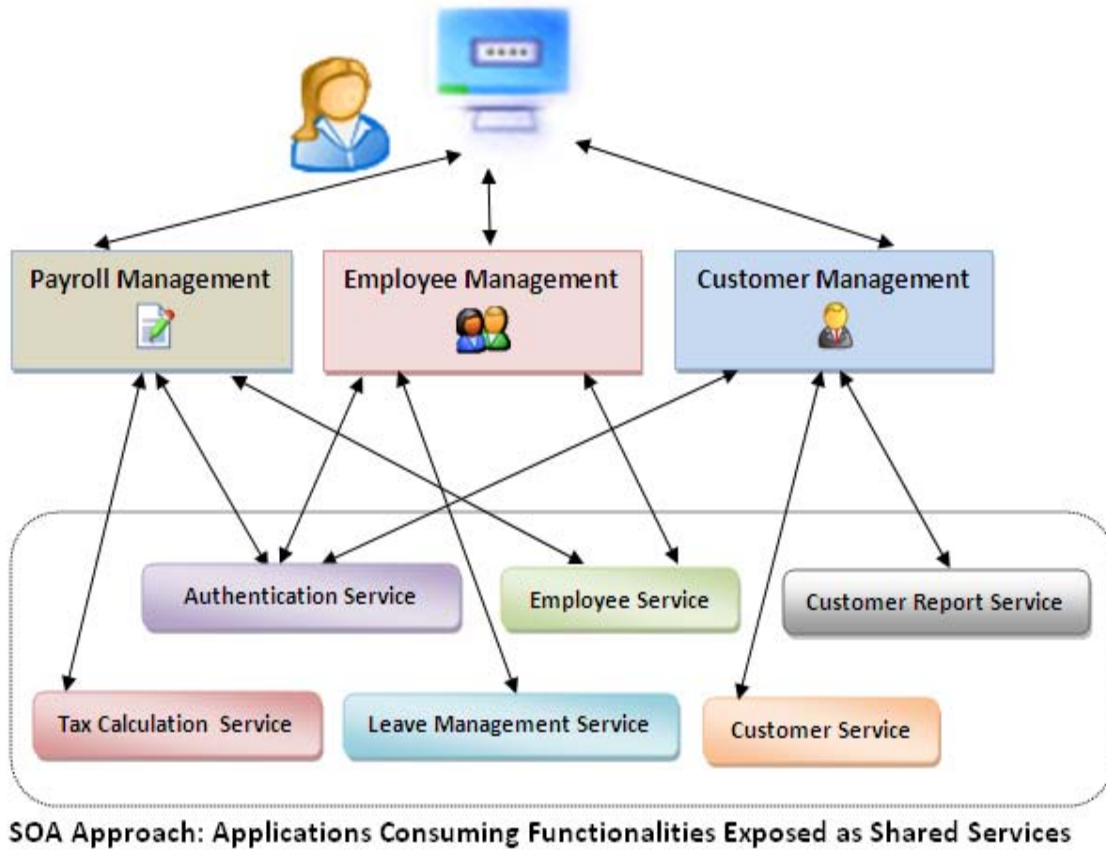


Figure 4

Service Oriented Architecture does not compete with 3-Tier architecture though; these two architectures offer two different approaches where one approach may be a best solution for a particular requirement, while other approach may not. While choosing the architecture, organization should access their requirement against some facts before drawing any decision.

Measure the SOA Benefits before Opting SOA Solution

Organization which are planning to invest in SOA solution should assess SOA benefits against their requirements on following grounds

- **What common functionalities are required across the applications?**
SOA offers reusability by exposing common functionalities as set of reusable components, which can be used in multiple projects. Reusability reduces redundancy, development effort, and risk. An example might be, an organization having multiple applications like HR, Payroll, Help Desk, Customer portal etc, where each application requires user authentication. Using SOA approach only one authentication component will be required and functionalities will be exposed as a reusable unit, which can be consumed by multiple projects hence, resulting in reducing redundancy, development, testing, and maintenance effort. While in the other approach, same component will have to develop per application resulting in increasing redundancy, development, testing, and maintenance effort.
- **Which changes in business process are more frequent?**
Business policies of organization keep changing due to the internal and external factors as board members drive partners, policies, and stakeholders but their business applications are not able to cope with such changes. SOA exposes business functionalities at a granularity, which can be easily recognized by the consumer and can be easily choreographed to support new changes.
- **What are the cross platform functionalities?**
SOA uses standard protocols such as XML, SOAP and WSDL for defining service interfaces which makes them consumable on any platform. For an example, a service is implemented in java and service interface is defined using standard protocols, can be consumed by .net applications as well.
- **Are you looking for business driven not technology driven solution?**
Never choose SOA just for the heck of technology; consider business drivers too. If you have well defined business drivers SOA will be a good candidate to choose.
- **Is there a need to reduce underlying IT- resources?**
For organizations having limited budget for underlying IT infrastructure and cannot afford costly resources in chunk, SOA will be a good option as this ensures optimum resource utilization and management. Using SOA, critical functionalities that require expensive hardware and software support can be exposed as a service running only on single server and client applications will have to just consume this service, eliminating the need of installing expensive software-hardware on every machine, thereby bringing down the IT infrastructure cost.
- **Is there a need for composite application? (Are you looking for the application needs to combine functionalities provided by multiple existing systems?)**
Integration is easily accomplished with service-oriented approach. While building Business-to-Business application the main concerned area is how to send sensitive information over the public network from one application to other. SOA provide robust, secure, and reliable solution by using secure channel and having authentication and data encryption in place.
- **Do you need consistency in business applications across the organization?**
SOA defines functionalities as reusable units, which can be used across multiple projects, offering great amount of consistency across the organization.

Appendix

Acronyms

SOA:	<i>Service Oriented Architecture</i>
B2B:	<i>Business to Business Application</i>
WF:	<i>Windows Work Flow Foundation</i>
WCF:	<i>Windows Communication Foundation</i>
XML:	<i>Extensible Markup Language</i>
HTML:	<i>Hyper Text Markup Language</i>
WSDL:	<i>Web Service Description Language</i>
SOAP:	<i>Simple Object Access Protocol</i>

About Proteans

Proteans is one of the leading outsourced Software Product Development Company and subsidiary of Norway based CAMO ASA (www.camogroup.com). We specialize in software product research and development services and help ISVs, SaaS providers, and Software Development Organizations of global 1000 companies worldwide to bring software products faster to market while reducing R&D costs. For more information, visit our website - www.proteans.com.

If you are considering implementing SOA strategy, call us today and we will be glad to offer you free assessment of your SOA needs.

US

Phone: +1-(410) 312 7625
Phone: +1-(410) 800-4679

UK

Phone: +44 (1708) 440 319

Norway

Phone: +47- 223 963 00

Sweden

Phone - +46 -707296481

India

Phone - +91 80 6618 6555