



Service Oriented Architecture

Go, change the world®

- **Service-oriented architecture (SOA)** is a method of software development that uses software components called services to **create business applications**. Each service provides a business capability, and services can also communicate with each other **across platforms and languages**.
- Service Oriented Architecture (SOA) is a specification and a methodology for providing platform and language independent services for use in distributed applications.
- A service is a repeatable task within a business process, and a business task is a composition of services. SOA describes **a message passing taxonomy for a component based architecture** that provides services to clients upon demand.
- Clients access a component that complies with SOA by passing a message containing metadata to be acted upon in a standard format.
- The component acts on that message and returns a response that the client then uses for its own purpose. A common example of a message is an **XML file transported over a network protocol such as SOAP**.



Service Oriented Architecture

Go, change the world®





Service Oriented Architecture

Go, change the world®

- This architecture does not contain executable links that require access to a specific API. The message presents data to the service, and the service responds.
- It is up to the client to determine if the service returned an appropriate result. An SOA is then seen as a method for creating an integrated process as a set of linked services. The component exposes itself as an “endpoint” (a term of art in SOA) to the client.
- The most commonly used message-passing format is an **Extensible Markup Language (XML)** document using **Simple Object Access Protocol (SOAP)**, but many more are possible, including **Web Services Description Language (WSDL)**, **Web Services Security (WSS)**, and **Business Process Execution Language for Web Services (WS-BPEL)**.
- WSDL is commonly used to describe the service interface, how to bind information, and the nature of the component’s service or endpoint.
- The **Service Component Definition Language (SCDL)** is used to define the service component that performs the service, providing the component service information that is not part of the Web service and that therefore wouldn’t be part of WSDL.



Service Oriented Architecture *Go, change the world®*

- Usually service providers and service consumers do not pass messages directly to each other. Implementations of SOA employ middleware software to play the role of transaction manager (or broker) and translator.
- **Middleware can discover and list available services**, as well as potential service consumers, often in the form of a registry, because SOA describes a distributed architecture security and trust services are built directly into many of these products to protect communication.
- **Middleware products also can be where the logic of business processes reside; they can be general purpose applications, industry-specific, private, or public services.**
- **Middleware services manage lookup requests. The Universal Description Discovery and Integration (UDDI) protocol is the one most commonly used to broadcast and discover available Web services, often passing data in the form of an Electronic Business using eXtensible Markup Language (ebXML) documents.**
- **Service consumers find a Web service in a broker registry and bind their service requests to that specific service; if the broker supports several Web services, it can bind to any of the ones that are useful.**



RV College of
Engineering®

Service Oriented Architecture

Go, change the world®

- The defining concepts of **Service-Oriented Architecture** vary from company to company, there are six key parameters that over reach the broad concept of Service-Oriented Architecture.

Core values include:

- Business value
- Strategic goals
- Intrinsic inter-operability
- Shared services
- Flexibility
- Evolutionary refinement

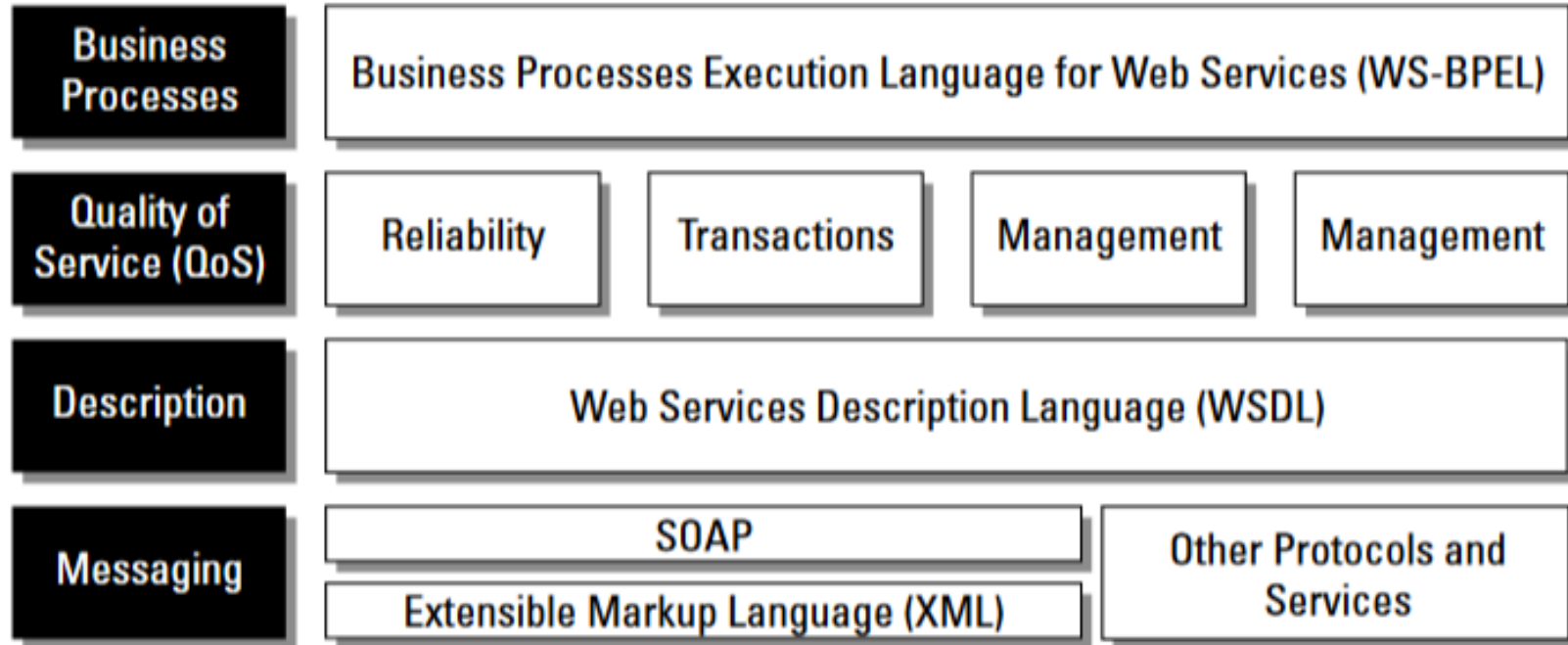


Service Oriented Architecture *Go, change the world®*





A protocol stack for SOA showing the relationship of each protocol to its function





Service Oriented Architecture *Go, change the world®*

- A protocol stack for an SOA architecture and how those different protocols execute the functions required in the Service Oriented Architecture.
- In the figure, the box labeled Other Services could include Common Object Request Broker Architecture (CORBA), Representational State Transfer (REST), Remote Procedure Calls (RPC).
- Distributed Common Object Model (DCOM), Jini, Data Distribution Service (DDS), Windows Communication Foundation (WCF), and other technologies and protocols. It is this flexibility and neutrality that makes SOA so singularly useful in designing complex applications.
- This architecture does not contain executable links that require access to a specific API. The message presents data to the service, and the service responds. It is up to the client to determine if the service returned an appropriate result.
- An SOA is then seen as a method for creating an integrated process as a set of linked services. The component exposes itself as an “endpoint” (a term of art in SOA) to the client.
- The most commonly used message-passing format is an Extensible Markup Language (XML) document using Simple Object Access Protocol (SOAP), but many more are possible, including WebServices Description Language (WSDL), Web Services Security (WSS), and Business Process Execution Language for Web Services (WS-BPEL). WSDL is commonly used to describe the service.



Service Oriented Architecture *Go, change the world®*

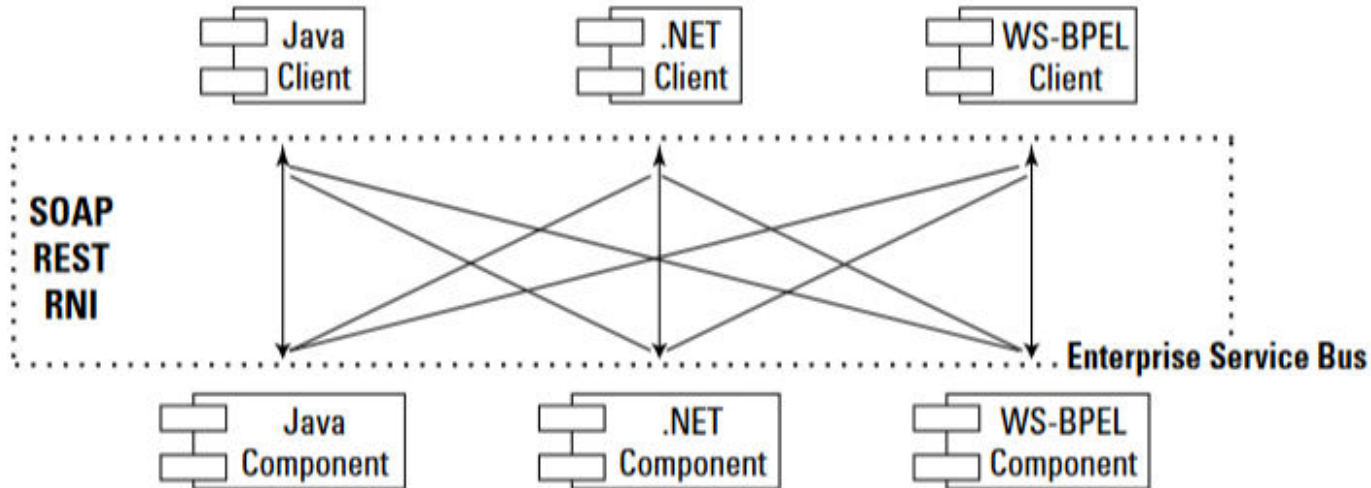
- Interface, how to bind information, and the nature of the component's service or endpoint. The Service Component Definition Language (SCDL) is used to define the service component that performs the service, providing the component service information that is not part of the Web service and that therefore wouldn't be part of WSDL.
- Note whatever protocol is used to negotiate a transaction, the formal definition of the transaction is referred to as the "contract." Indeed, the notion of a contract implies a certain level of service that is available to clients and that may be part of any paid service in SOA.
- Figure shows a protocol stack for an SOA architecture and how those different protocols execute the functions required in the Service Oriented Architecture.
- SOA provides the framework needed to allow clients of any type to engage in a request response mechanism with a service. The specification of the manner in which messages are passed in SOA.
- SOA requires the use of an orchestrator or broker service to ensure that messages are correctly transacted. SOA makes no other demands on either the client (consumer) or the components (provider) of the service; it is concerned only with the interface or action boundary between the two.

SOA-Client Construction

Go, change the world®

SOA allows for different component and client construction, as well as access to each using different protocols.

Clients (Consumers)



Services (Providers)



SOA- Choreography

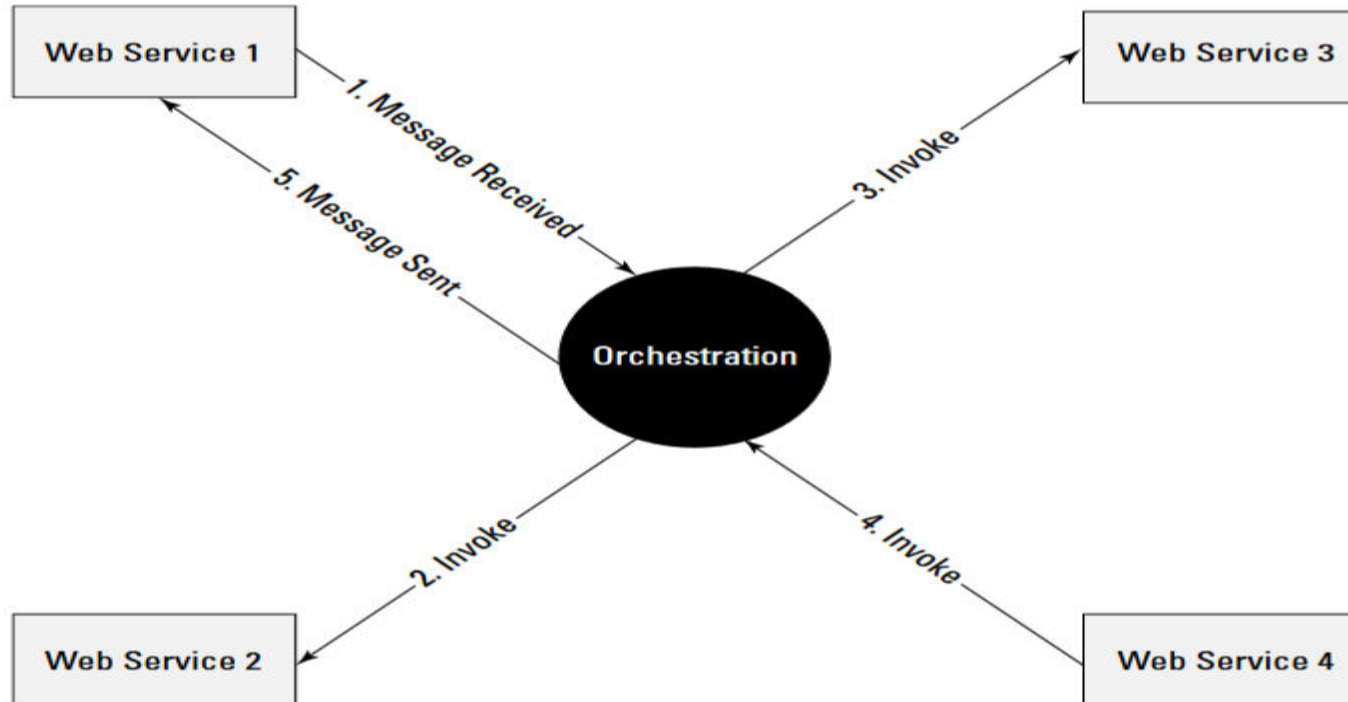
- Figure shows how components of different types can communicate using different protocols as part of SOA.
- When you combine Web services to create business processes, the integration must be managed.
- Two main methods are used to combine Web services: orchestration and choreography.
- In orchestration, a middleware service centrally coordinates all the different Web service operations, and all services send messages and receive messages from the orchestrator.
- A compound business process that uses **choreography has no central coordination function**. In choreography, each Web service that is part of a business process is aware of when to process a message and with what client or component it needs to interact with.
- **Choreography** is a collaborative effort where the logic of the business process is pushed out to the members who are responsible for determining which operations to execute and when to execute them, the structure of the messages to be passed and their timing, and other factors.



Orchestration

Go, change the world®

An orchestrated business process uses a central controlling service or element, referred to as the orchestrator, conductor, or less frequently, the coordinator.





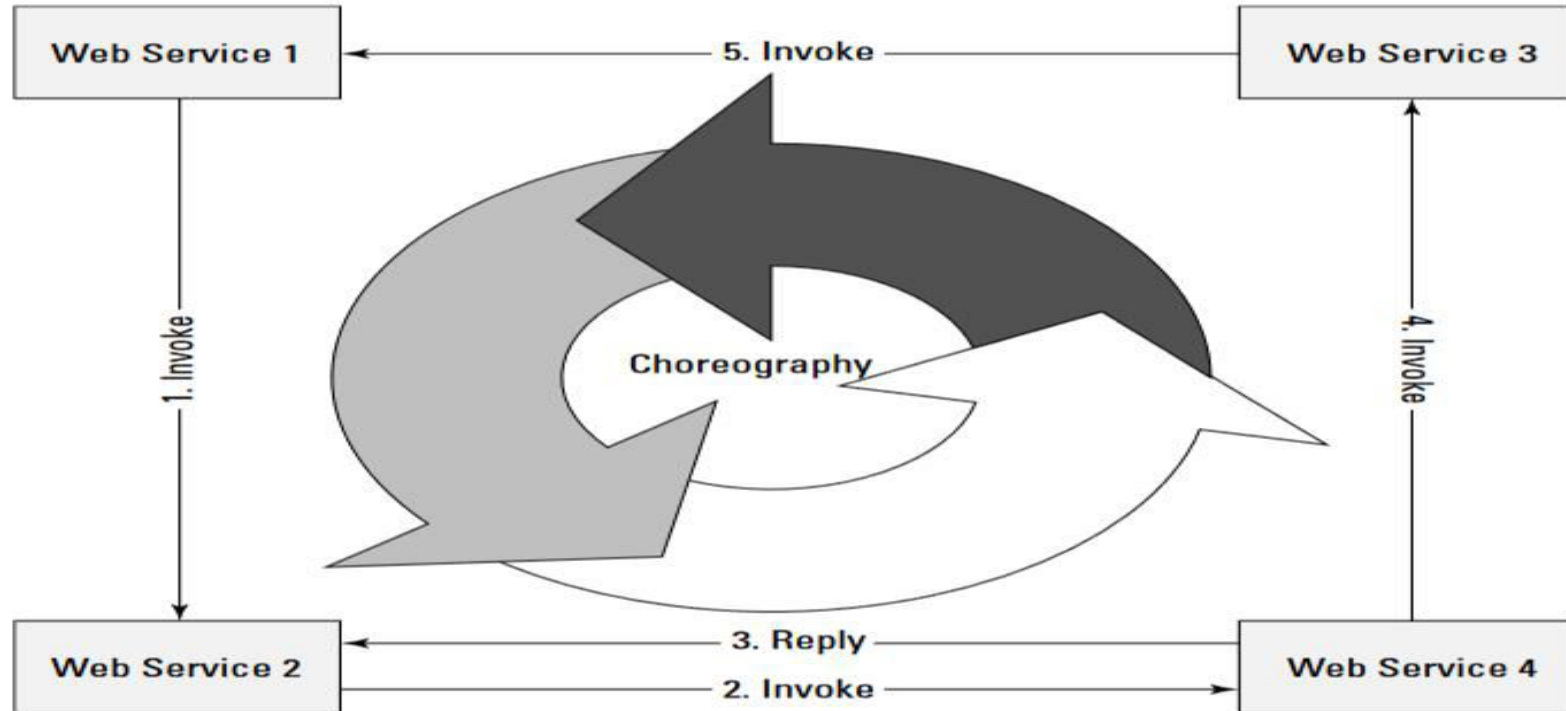
SOA- Choreography

- Most mature SOA implementations favour orchestration over choreography for a number of reasons.
- With orchestration a single central service manages the various processes, and changes to the business logic can be made in that one location.
- The integration of Web services into the architecture is easier than with choreography because these services don't need to know anything about the business process.
- Centralizing the business logic also makes it easier to put error handling mechanisms in place and to account for, manage, and analyze events that occur outside the business process that relate to a part of the process.
- Event handling is part of event-driven SOA or SOA 2.0, which extends Service Oriented Architecture to include both random and scheduled events that are triggered by a business process outside of a business process.
- One way of performing orchestration is through the use of an Enterprise Service Bus or ESB. An ESB provides a middleware software layer for event management with a messaging infrastructure.

SOA- Choreography

Go, change the world®

With choreography, business process execution is a cooperative affair.





- Event-driven SOA or SOA 2.0 is an extension of the Service Oriented Architecture to respond to events that occur as a result of business processes or perhaps cause and influence a business process.
- For example, in a business process, sales at a certain Web site are processed. If the business process recognizes the rate at which sales are occurring, it could perform an analysis to determine what events might influence the buying decision.
- This is the sort of analysis that event-driven SOA is meant to address. SOA 2.0 can allow low-level events to trigger a business process, correlate events with information contained in the SOA design, inhibit a business process if the appropriate events don't appear, or invoke a reaction or response based on a trigger.
- To perform these tasks in SOA 2.0, a **Causal Vector Engine (CVE)** with some built-in artificial intelligence must be added to the SOA design.
- Events are analyzed in terms of event sequences, event relationships, and event timing to establish whether a certain condition has occurred.
- The CVE then determines how to react to the condition using a set of rules that are built into the system. Many CVE systems display events in a console in different contexts so that an observer can



- The CVE then determines how to react to the condition using a set of rules that are built into the system. Many CVE systems display events in a console in different contexts so that an observer can analyze the display and take appropriate actions.
- A CVE application may include the ability to query event data in the same way that a stock ticker or trading application can query trading data.
- The CVE application provides the same kind of heartbeat and correlation functionality that a stock trading application does.
- From the standpoint of the service requestor or consumer (client), the client simply needs to know the form required to initiate the action of the provider (service) and how to interpret the results returned from the service provider.
- The nature of the component's processing is unknown, the location where the processing is done is unknown, and the various operating systems and applications involved are unknown.



Event-driven SOA or SOA 2.0

Go, change the world®

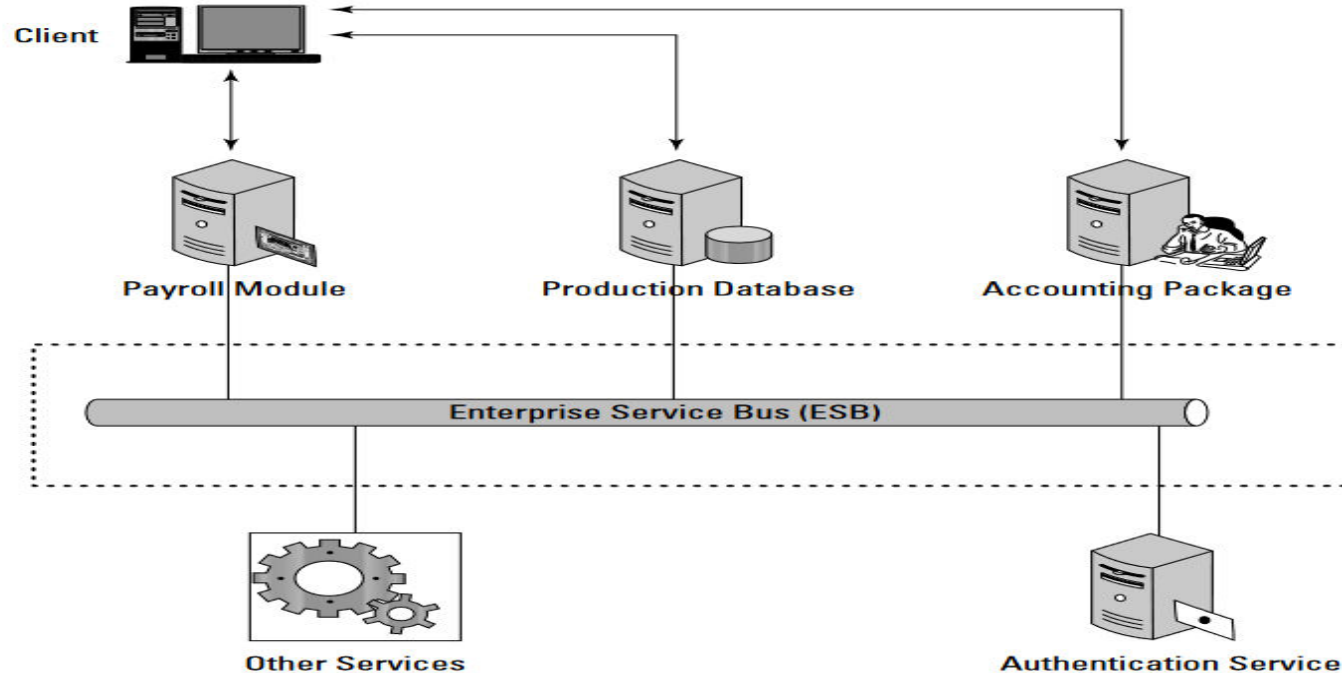
- The client is responsible for validating that the service returned the results that were expected. The SOA component is essentially a black box to the client.
- SOA makes no demands of the component other than to conform to the rules of a standard end point.
- This level of abstraction offers operational advantages to Web service providers in that components can be continually upgraded, replaced, or moved to improve efficiencies without disrupting the clients that depend on those services, and the Quality of Service for that service can be accurately measured and delivered.
- SOA provides the rules so each application can access the authentication module in its own way, as required.
- What you gain with SOA is the ability to add significant capabilities with a fraction of the cost or effort and to federate applications if you desire.
- What you lose with SOA is the ability to perform fundamental customization of the service itself when that service is provided by a third party.



The Enterprise Service Bus

Go, change the world®

An SOA application of a shared logon or Authentication module





The Enterprise Service Bus

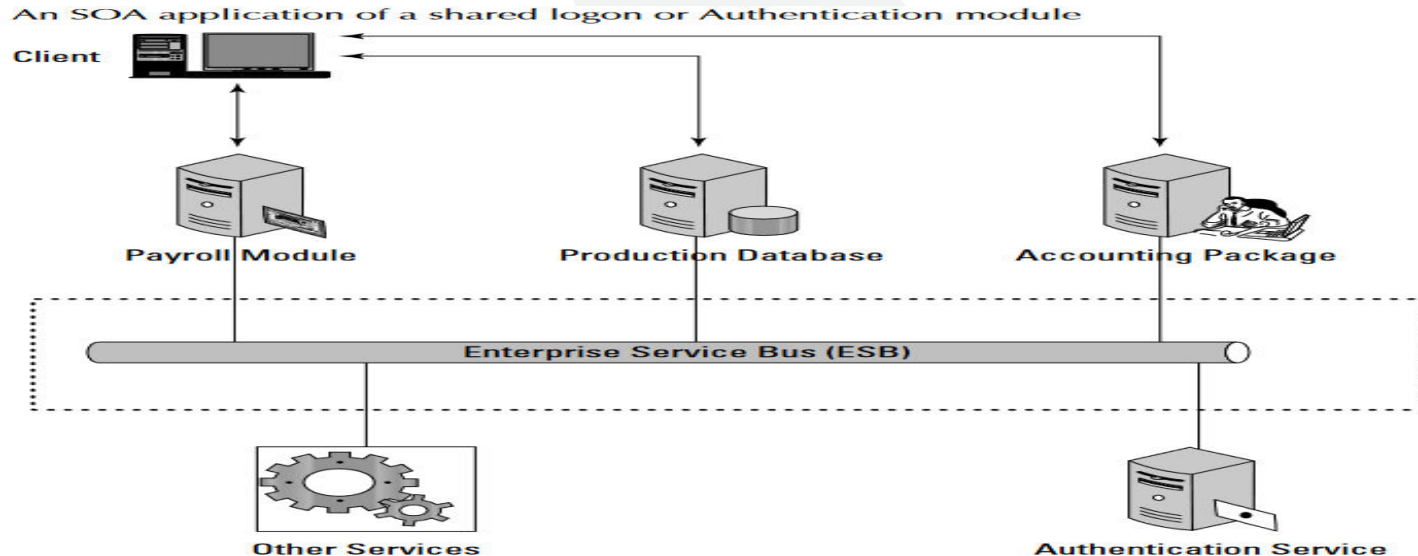
Go, change the world®

- An ESB is not a physical bus in the sense of a network; rather, it is an architectural pattern comprised of a set of network services that manage transactions in a Service Oriented Architecture.
- An ESB as a set of services that separate clients from components on transactional basis and that the use of the word bus in the name indicates a high degree of connectivity or fabric quality to the system; that is, the system is loosely coupled.
- Messages flow from client to component through the ESB, which manages these transactions, even though the location of the services comprising the ESB may vary widely.
- An ESB is necessary but not essential to a Service Oriented Architecture because typical business processes can span a vast number of messages and events, and distributed processing is an inherently unreliable method of transport.
- An ESB therefore plays the role of a transaction broker in SOA, ensuring that messages get to where they where supposed to go and are acted upon properly.

The Enterprise Service Bus

Go, change the world®

- The service bus performs the function of mediation: message translation, registration, routing, logging, auditing, and managing transactional integrity.
- Transactional integrity is similar to ACID in a database system—atomicity, consistency, isolation, and durability, the essence of which is that transactions succeed or they fail and are rolled back.





The Enterprise Service Bus

Go, change the world®

- An ESB may be part of a network operating system or may be implemented using a set of middleware products. An ESB creates a virtual environment layered on top of an enterprise messaging system where services are advertised and accessed.
- Think of an ESB as a message transaction system. IBM's WebSphere ESB 7.0 is an ESB based on open standards such as Java EE, EJB, WS-Addressing, WS-Policy, and Kerberos security, and it runs on the WebSphere Application Server.
- It is interoperable with Open SCA. WebSphere ESB contains both a Service Federation Management tool and an integrated Registry and Repository function.



Typical features are found in ESBs, among others:

- Monitoring services aid in managing events.
- Process management services manage message transactions.
- Data repositories or registries store business logic and aid in governance of business processes.
- Data services pass messages between clients and services.
- Data abstraction services translate messages from one format to another, as required.
- Governance is a service that monitors compliance of your operations with governmental regulation, which can vary from state to state and from country to country.
- Security services validate clients and services and allow messages to pass from one to the other.



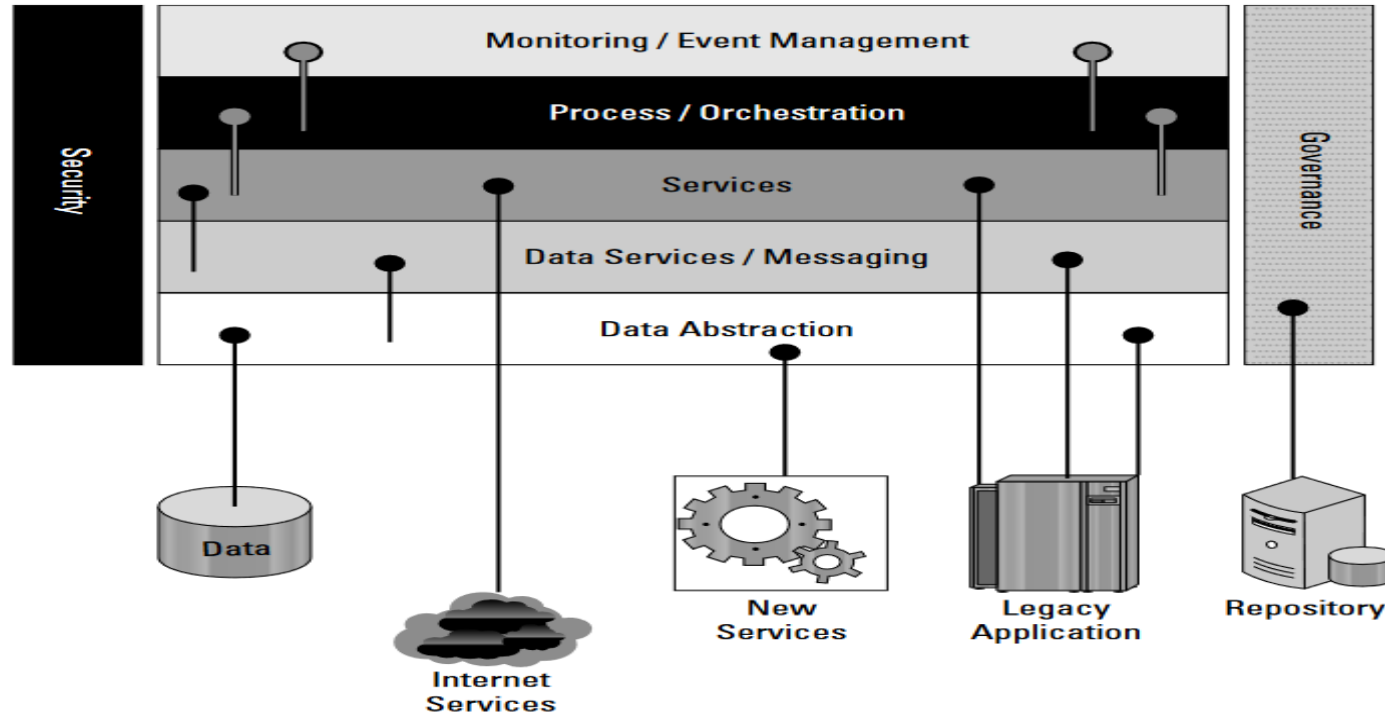
The Enterprise Service Bus

Go, change the world®

- The difference between a repository and a registry in the context of a Service Oriented Architecture is subtle. Repositories and registries are both data stores, but a repository stores references to the components of the SOA, their source code, and linking information that are used to provide SOA services.
- An SOA registry contains references to rules, descriptions, and definitions of the services that is, the metadata of the components.
- A repository serves the role that a name server does in a network operating system infrastructure, while the registry plays the role of a directory service (domain).
- The service broker uses the rules contained in the SOA registry to perform its function as translator and delivery agent. For developers, the registry serves as the central location to store component descriptions that allow composite applications to be created and the place in which services may be published for general use.
- These services in an SOA also include the provider interfaces and standard sets of network protocols that were mentioned previously. Developers may also choose to create a Business Process Orchestration module to coordinate the access and transactional integrity of multiple business applications that integrate into a larger platform,

SOA-Network Infrastructure *Go, change the world®*

This figure shows a network services model infrastructure for an SOA, which is based on the SOA meta-model of the Linthicum Group, 2007. A slightly different version of this diagram appears in *Networking Bible* by Barrie Sosinsky, Wiley, 2009.





Service Catalogs

- Finding any particular service and locating the service's requirement in a large SOA implementation can involve a large amount of network system overhead.
- To aid in locating services, SOA infrastructure often includes a catalog service. This service stores information on the following, among other things:
 - What services are available, both internal and external
 - How to use a service
 - Which applications are related to a particular service (dependencies)
 - How services relate to one another
 - Who owns the service and how a service is modified
 - The event history of a service, including service levels, outages, and so on
 - The nature of service contracts



Service Catalogs

- Service catalogs are dynamic and under constant modification. Catalog servers have these features:
- They can be standalone catalog servers serving a single site.
- They serve the role of a global catalog service where two or more catalog servers are merged to include several sites. A global service usually requires some sort of synchronization or update to maintain a unified data store across the servers involved.
- They can be part of a federated catalog service where two or more global catalog servers have access to one another's information through a trusted query relationship.
- Catalog services have an enormous impact on large system performance and eventually become essential as a SOA internetwork system grows.
- An internetwork is a network that is constructed through the consolidation of separate networks, in the same manner that the Internet has been built.