Oracle SOA Suite – Enterprise Service Bus
Integrate Services and Events into Flexible Enterprise Solutions
EXECUTIVE OVERVIEW
Changing markets, increasing competitive pressures and evolving customer needs are placing greater pressure on IT to deliver greater flexibility and speed. In response to these challenges, leading companies are adopting Service-Oriented Architecture (SOA) as a means of delivering on these requirements by overcoming the complexity of their application and IT environments. SOA represents a fundamental shift in the way new applications are designed, developed, and integrated with legacy business applications, and facilitates the development of enterprise applications as modular business services that can be easily integrated and reused.

Within a SOA environment messaging is a critical requirement. Oracle Enterprise Service Bus (ESB) is the foundation for delivering enterprise solutions based on Service-Oriented and Event Driven computing. At its core, it is a loosely coupled application framework that provides businesses with increased flexibility, reusability and overall responsiveness in a distributed, heterogeneous, message-oriented environment.

Oracle SOA Suite is a standards-based best of breed suite that enables you to build Service-Oriented Application and deploy them to your choice of middleware platform. It consists of (i) a Integrated Service Environment (ISE) to develop services; (ii) a multi-protocol Enterprise Service Bus (ESB) to integrate applications; (iii) a services registry for discovering and managing the lifecycle of services; (iv) a BPEL-based orchestration engine to tie services into business processes; (v) a business rules engine to enable business policies to be captured and automated; (vi) a Web Services management and security solution to enforce authentication and authorization policies on services and to monitor services and processes for compliance to SLAs; (vii) a Business Activity Monitoring (BAM) solution to gain real-time visibility into business entities and their interactions, and a means to enable automated actions to be performed and (viii) an Enterprise Portal for employees, customers and partners to access content, access relevance performance metrics, collaborate and take actions via interaction with business processes.

Oracle SOA Suite can help you achieve greater organizational flexibility better than any other solution in the market. It can reduce your costs and middleware complexity better than any other solution. Finally, it can help you to achieve the best total value of opportunity.
INTRODUCTION – ORACLE FUSION ARCHITECTURE

Today, every organization is faced with the need to predict changes in the global business environment, to rapidly respond to competitors, and to best exploit organizational assets to prepare for growth. Your enterprise application infrastructure can either help you meet these business imperatives or it can impede your ability to change. To help you, your infrastructure must:

- Improve your ability to predict and respond to change
- Enhance organizational productivity
- Simplify your information technology environment
- Leverage existing investments

In order to deliver on these requirements and overcome the complexity of their IT environments, leading companies are adopting Service-Oriented Architecture (SOA). SOA represents a fundamental shift in the way new applications are designed, developed, and integrated with legacy business applications, and facilitates the development of enterprise applications as modular business services that can be easily integrated and reused.

A core capability within a SOA environment is messaging. Oracle ESB, a component of Oracle SOA Suite, provides messaging, routing and transformation capabilities that enables services to be easily integrated at development time and runtime.

Oracle SOA Suite is a standards-based best of breed suite that enables you to build Service-Oriented Application and deploy them to your choice of middleware platform. It consists of:

1. An Integrated Service Environment (ISE) to develop services
2. A multi-protocol Enterprise Service Bus (ESB) to integrate applications
3. A Services Registry for discovering and managing the lifecycle of services
4. A BPEL-based orchestration engine to tie services into business processes
5. A Business Rules Engine to enable business policies to be captured and automated
6. Web Services management and security solution to enforce authentication and authorization policies on services and to monitor services and processes for compliance to SLAs
7. A Business Activity Monitoring (BAM) solution to gain real-time visibility into business entities and their interactions, and a means to enable automated actions to be performed
8. An Enterprise Portal for employees, customers and partners to access content, access relevance performance metrics, collaborate and take actions via interaction with business processes.

This paper will show how Oracle ESB can be used to integrate Services in Service Oriented Architectures.
ORACLE ESB OVERVIEW

Definition
Oracle ESB is a standards based infrastructure component of the Oracle SOA Suite delivering loosely coupled data and enterprise application integration that is (i) fast, (ii) secure, (iii) reliable and (iv) highly available. Oracle ESB features a multi-protocol message bus with centralized monitoring management of distributed services where all services are exposed as standard web services using WSDL. Oracle ESB contains message flows utilizing adapters, transformations and routing rules to distribute data throughout and beyond the enterprise.

Components
The Oracle ESB system is made up of the following service components: (i) designer, (ii) repository and registry, (iii) runtime server, and (iv) management and monitoring console. The component details are described below.

Service Designer
Oracle ESB in either the JDeveloper or Eclipse Integrated Service Environment includes an ESB tools for modeling JCA adapters, XSLT transformations and routing rules with XPATH based filter expressions. The same ISE includes editors for building: XSD schemas, Java programs, EJ Bs, JSP pages, WSDL files and a host of other J2EE related service componentry.

Service Repository and Registry
ESB project metadata artifacts such as schemas, transformations and routing rules are stored into a common service repository and service WSDL files are optionally registered into the Oracle SOA Suite UDDI component. The repository is configurable and provides granular import/export of ESB service artifacts for decoupled life cycle management.

Service Runtime Engine
Oracle ESB includes a service engine or server that is deployed as a J2EE application on industry leading applications servers such as Oracle OC4J, JBOSS, BEA WebLogic and IBM WebSphere. The ESB server provides the primary communication for the EM ESB Grid Control service console and accesses metadata from the shared service metadata repository.

Service Management and Monitoring
The Oracle Enterprise Manager and ESB Grid Control console is used to manage ESB systems and clusters, monitor services, configure routing rules, build domain value maps and monitor and manage runtime queues and services. The consoles routing rules editor uses an expression builder for creating filter expressions that enables dynamic content based routing. The console also offers simplified service distribution by modifying local/distributed and synchronous/asynchronous settings. Oracle AS EM provides management of the server and it J2EE infrastructure components including start, stop, auditing and messaging services. Figure 1 shows the Oracle AS EM JMS management page.

*Oracle's ESB and Business Integration offering is an important component of Oracle® Fusion Middleware and helps customers increase flexibility, reusability and overall responsiveness in a distributed, heterogeneous, service-oriented environment. Oracle remains committed to supporting open standards and to delivering customers and partners best-in-class components for developing an SOA infrastructure.*

- Amlan Debnath, Vice President, Server Technologies, Oracle
Figure 1: Oracle AS EM JMS

**ORACLE ESB ARCHITECTURE**

Oracle ESB provides a hierarchical separation of services into logical groupings called Systems and Service Groups. These groupings improve distribution of services across clusters and allow for ease of life cycle management based on departmental or functional areas. Table 1 defines and illustrates the hierarchical relations of services and groupings.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM</td>
<td>Top level grouping</td>
<td>OracleApps</td>
</tr>
<tr>
<td>SERVICE GROUP</td>
<td>Service sub grouping</td>
<td>Customer</td>
</tr>
<tr>
<td>SERVICE</td>
<td>Adapter, Routing, SOAP</td>
<td>Adapter: CustomerEvent Routing: CustomerRS SOAP: CustomerCreditCheck</td>
</tr>
</tbody>
</table>

Table 1: Service Organization Illustration
Figure 2 illustrates the relationship and data flow between the ISE, ESB Server, Metadata Repository and Enterprise Manager monitor in a single Oracle ESB installation instance.

The runtime engine supports multiple protocol bindings including HTTP/SOAP, JMS, JCA, RMI, WSIF and Java that ensure guaranteed, reliable message delivery using synchronous/asyncronous, request/reply or publish/subscribe models. The engine either commits or rolls back existing synchronous transactions and uses a dead letter queue mechanism with resubmit semantics for asynchronous transactions. The stateless message based engine delivers distributed high performance throughput but as a result does not natively support certain interactions such as asynchronous request/reply. The engine also provides native support for Oracle Enterprise Messaging Services various qualities of service and levels of message persistence including database, files and in-memory. When JMS is configured for database persistence, Oracle AQ is used thus exploiting all of the benefits of Oracle Database Grid management for scalability, reliability and availability. ESB engine clustering (see figure 3) enables centralized metadata management with a fully distributed runtime service topology that is resilient to metadata repository outages.
INTEGRATION USE CASE

To illustrate a common integration scenario, an integration engineer uses Oracle ESB for real-time synchronization of customer data from an Oracle Application Suite system to PeopleSoft and Siebel CRM. Figure 4 shows the completed integration scenario in the EM ESB control illustrating the service hierarchy and relationship diagram. The routing services depicted by a vertical object contain a “Routing Rules” tab for dynamically configuring schemas, rules and filter expressions for outbound service invocations.

![Oracle ESB Relationship Diagram](image)

Figure 4: Oracle ESB Relationship Diagram

The engineer starts by using the JDeveloper Integrated Service Environment to create 4 new ESB systems: one for each enterprise system and 1 specifically for canonical services. Next, an ESB service group for customer metadata is created inside each of these systems to accommodate future expansion to other modules such as Financials or ERP.

Build the Canonical Service

Using JDeveloper, the engineer completes the following steps to build the canonical model:

1. Create a project and import the adopted canonical schema.
2. Create an outbound canonical routing service and select the schema element. (Fig. 5)
3. Register the project with the server.
Building the Inbound Adapter Service

The integration engineer works with the enterprise applications expert to access and define the inputs and outputs for the respective adapters. The following steps are required for the inbound OracleApps adapter:

1) Create a project and import the OracleApps adapter schema.

2) Invoke the OracleApps adapter wizard (Fig. 6) to model extraction of the required customer data objects to the inbound schema.

3) Create routing rules to link the incoming event to the outbound taxonomical routing service created above.

4) Invoke the transformation mapper GUI to create a map from the inbound customer schema to the canonical schema in the outbound canonical service.

5) Register the project with the server.

6) When the project is registered, a new adapter routing service is automatically created that contains the transformation and routing rules to invoke the outbound canonical routing service.
Building the Outbound Adapter Service

The engineer builds the outbound Siebel and PeopleSoft. The steps include:

1) Create JDeveloper project and import outbound customer schemas.
2) Use adapter wizard or SOAP services to create outbound adapters to the systems.
3) Use transformation mapper GUI to build mapping from the canonical routing service to the outbound adapter or SOAP service.
4) Create routing rules by selecting inbound canonical routing service to invoke the outbound adapter or SOAP service.
5) Register the project with the server.
6) Figure 7 shows the JDeveloper customer project hierarchy and mapper GUI.
DETAILED FEATURE FUNCTIONALITY

Connectivity
ESB Connectivity is characterized by how the data is accessed and what rules and infrastructure are involved in unlocking the data. Below are descriptions for the types of connectivity services Oracle ESB provides.

- **Adapters** - Oracle Adapters provide connectivity to virtually any data source in your enterprise by providing bi-directional, real-time connectivity using open standards such as JCA, WSDL, XML, JMS, Web Services and WSIF. Oracle Adapters provide service interfaces that enable fast SOA access to your technology investments.

- **Application Events** - Oracle ESB is tightly coupled with Oracle Applications, PeopleSoft and JD Edwards using a business events system that enables visibility into their respective systems metadata and event structure. Oracle ESB is the main access point for applications data being sent to the entire Oracle Business Integration SOA platform including BPEL, PM, BAM and Oracle B2B.

- **Metadata Services** - Oracle Metadata Services or MDS manages all types of metadata across the Oracle Application Server product line including adapters, transformations, and routing and business rules. This common metadata model is key to unlocking the data in your enterprise and enabling integrated end-to-end service modeling, monitoring and management throughout the Oracle SOA stack.

- **Security** - Protecting critical organizational data is an important requirement to execute effectively in a service oriented architecture. Oracle ESB uses Oracle Identity Managements integrated platform security infrastructure including authorization, authentication, encryption and decryption to move data securely across the enterprise.

Document Enrichment
Enrichment generally describes any document modifications required to tailor the data for its outbound consumers. Enrichment can be illustrated either as a single step transformation or multiple actions that implement specialized transformations, lookups, rules or system indexing functionality. Oracle ESB provides the following enrichment functionality.

- **Document Transformations** - Oracle ESB includes standards based data mapper functionality within the Oracle JDeveloper or Eclipse modeling environment to create XSLT transformation services for reuse across the enterprise. The auto-mapping feature remembers and reuses common mappings from previous transformations. The engine supports multiple XSLT and XQUERY providers.

- **Business Rules** - Oracle Business Rules engine and Rules Editor allow administrators to de-couple business logic from core service logic. The Rules Editor empowers business analysts to model complex custom services implementing routing rules or document enrichment.
• **System Cross References** - System cross-referencing provides dynamic indexing of keys from disparate systems to automatically map a document such as an order in one system to the matching order-id in another.

• **Domain Value Mappings** - Oracle ESB has robust support to easily map domain values such as state or currency codes from one system to another. The user interface tool allows many to one mapping with default values that are well suited for canonical schema management use cases.

**Document Routing**
Routing is the final stage of the Connect, Enrich and Route pattern. The basic functionality of content routing is augmented by other requirements such as messaging interoperability, distributed message propagation and flexible, scalable and highly available service deployments.

• **Content Based Routing** - Oracle ESB exposes content-based routing (CBR) rules during design time that can be modified at runtime to adjust application efficiency. It supports a variety of rules engines including Oracle Business Rules and external providers. Content filtering can be implemented in messaging systems such as JMS using configurable filter based subscriptions and header message selectors. CBR services can be deployed with Oracle JDeveloper using built-in Router services, Java, PL/SQL or Oracle BPEL PM.

• **Multi Protocol Messaging Bus** - Oracle ESB provides a flexible real-time enterprise backbone supporting multiple protocols where all services have typed payloads using the Web Services Definition Language (WSDL). It enables special in-memory optimization for service calls within the same virtual machine.

• **Open Standards and Interoperability** - The Oracle SOA platform is committed to supporting open standards and 3rd party products from the operating system to the application server and JMS messaging space. Open standards and 3rd party interoperability enable you to leverage your existing infrastructure and increase overall productivity. JMS, SOAP, WS-Addressing, WS-Security, JCA, WSIF, JBI, BPEL, JBOSS, JDBC, HTTP, FTP are examples of the platforms standards support which is constantly evolving.

• **Oracle Enterprise Messaging Service** - Oracle ESB uses the Oracle Enterprise Messaging Service (OEMS) messaging infrastructure for managing fast, scalable, guaranteed service for point-to-point and publish/subscribe patterns. OEMS provides interoperability with many 3rd party messaging providers and delivers the following JMS qualities of service (i) Database persistence (ii) File persistence and (iii) In-Memory. It also provides distributed message propagation to serve messaging gateway uses.

• **Scalable and Highly Available Services** - Oracle ESB and the SOA platform leverage the Oracle Application Server GRID clustering management features. This proven infrastructure technology ensures applications are highly available and scalable to meet all of your redundancy and performance requirements.
WHY ORACLE?

Oracle SOA Suite is the only comprehensive and integrated SOA suite in the industry. While other vendors claim to have similar platforms, Oracle SOA Suite provides several unique differentiators over other products.

Realize Greater Organizational Flexibility

Oracle SOA Suite can help you achieve greater organizational flexibility better than any other solution in the market. Oracle ESB facilitates this with the following:

- **Standards Based Integration** – Oracle ESB utilizes standards that reduce barriers to integrating systems enabling better decision-making, and new products to be rolled out quicker and easier than before.

- **Business Resiliency** – Oracle ESB simplifies change by enabling configuration-based changes to service artifacts such as schemas, transformations, routing rules and content based routing filter expressions without requiring costly redeployment.

- **Business Visibility** – Oracle ESB increases business visibility into enterprise systems by exposing registering all adapter and routing services as standard Web Services via WSDL into UDDI service registries.

Oracle SOA Suite supports:

a range of application servers including IBM WebSphere and JBoss; leading business rules engines such as Ilog Jrules and Corticon; any LDAP V3 compliant directory such as Active Directory, iPlanet and Novell; leading messaging services such as IBM MQ, SonicMQ, Tibco as well as Oracle AQ.

Eliminate Middleware Complexity

Oracle SOA Suite can reduce your costs and middleware complexity better than any solution available from any other vendor. It is the industry’s only SOA Suite technically engineered to be a single product. Oracle SOA Suite differs from other market solutions in four key areas:

- **Single Development Framework** – Oracle SOA Suite is the only SOA suite that provides a single integrated design time environment to develop enterprise applications, to compose Web services, to create enterprise portals, and to orchestrate business processes. You learn one tool to target the entire platform.

- **Single Deployment Architecture** – Oracle SOA Suite is the only SOA suite that provides a common architecture for scalability, availability, workload distribution, resource management, security, and metadata management. You spend less time integrating your middleware infrastructure.

- **Single Management Architecture** – Oracle SOA Suite is the only SOA suite that has a common identity management and systems management architecture. You monitor and manage users and systems centrally, lowering cost and improving security.

- **Single Metadata Management System** – Oracle SOA Suite is the only SOA suite that leverages a common metadata management system across all components, speeding up application development and leading to more maintainable applications.

- **Easy to Adopt** – All of the SOA Suite components are built upon and support industry standards, to ensure that they can be incrementally adopted and easily integrated into an organization’s existing information technology infrastructure.
Oracle SOA Suite integrates seamlessly into your existing IT environment. This “hot-pluggable” architecture is shown in Figure 8.

![Oracle SOA Suite, Hot-Pluggable Architecture](image)

**Figure 8: Oracle SOA Suite, Hot-Pluggable Architecture**

**Achieve Best Total Value of Opportunity**

Oracle SOA Suite can help you achieve the best total value of opportunity by means of three reduction strategies:

- **Develop and Deploy Applications Faster** – As the market’s only integrated SOA Suite, Oracle SOA Suite greatly reduces the overall cost of architecting, developing, deploying, and managing applications. Applications are built faster, they can be put into production sooner, and their associated benefits can be realized sooner. You save money, and you can more quickly reassign software engineers and R&D funds to other projects.

- **Reduce Application Deployment Costs** – SOA Suite is the only SOA Suite designed to leverage grid computing to lower costs by deploying enterprise applications on modular, low-cost hardware and storage.

- **Reduce Maintenance and Management Costs** – Oracle SOA Suite lowers management costs by automating software provisioning across groups of systems and by centralizing systems monitoring and administration. It lowers security administration costs by centralizing identity and access management.

Oracle SOA Suite can help you achieve the best total value of opportunity by reducing your development costs, by reducing your deployment costs, and by reducing ongoing maintenance costs, thus providing you with the fastest return on your investment.
CUSTOMER PROOFPOINTS

Deutsche Post
Oracle has partnered with Deutsche Post World Net, a leader in the fast growing industry of mail, communication, parcel, express, logistics and financial services markets both in Europe and worldwide, to build a world class Service Oriented Architecture that meets the integration needs of their multifaceted business environment. By utilizing Oracle's ESB technology and working together with Oracle to build a Java Business Integration (JBI) container based the JSR 208 open standard, Deutsche Post will be prepared to manage their highly complex service-oriented business processes in an SOA framework.

CONCLUSION
Using proprietary, one-off custom solutions to integrate enterprise applications requires specific skill sets and domain knowledge that are often hard to find. Most custom solutions are not metadata-driven so any changes to the underlying systems require reprogramming which is difficult to maintain and labor intensive to roll out to the end customer. Oracle ESB solves this by providing real-time, bi-directional, widespread connectivity to various backend applications. Its comprehensive enrichment and data distribution features enable fast document manipulation and delivery of data to multiple destinations across many local and remote topologies. The pervasive monitoring and management functionality enables real-time visibility and quick responses to problems that previously may have taken precious hours to surface and diagnose.