Oracle E-Business Suite Adapter

Integration Methodology

Author
Neeraj Chauhan
(Manager – Product Management, Oracle E-Business Suite, ATG)
# Table of Content

**Introduction** ................................................................................................................................. 3  
**Overview** ..................................................................................................................................... 4  
  Why Integration ............................................................................................................................... 4  
  Why Adapters? ............................................................................................................................... 4  
  Oracle E-Business Suite Adapter ................................................................................................. 4  
**Oracle E-Business Suite Adapter Integration Methodology** ....................................................... 5  
  DEFINE Phase ............................................................................................................................... 6  
    Scope – Objectives – Goals ......................................................................................................... 6  
    Prototyping The Possibilities .................................................................................................... 6  
    Checkpoints at DEFINE phase ............................................................................................... 7  
  DESIGN Phase ............................................................................................................................... 8  
    Identifying The End Points ....................................................................................................... 8  
    Identifying The Direction Of Transactions ........................................................................... 9  
    Architecting The Solution ....................................................................................................... 9  
    Checkpoints at DESIGN phase ............................................................................................. 9  
  DEVELOP Phase ............................................................................................................................. 10  
    Develop The Solution ............................................................................................................. 10  
    Test-Run the solution ............................................................................................................. 10  
    Checkpoints at DEVELOP phase .......................................................................................... 11  
  Deploy Phase .................................................................................................................................. 12  
    Deploy the solution .................................................................................................................. 12  
    Checkpoints at DEPLOY phase .............................................................................................. 12  
**Best Practices And Recommendations** ....................................................................................... 13  
  Securing Integration ..................................................................................................................... 13  
  Enable Function Security ............................................................................................................. 13  
  Secured connection between Oracle E-Business Suite and Fusion Middleware .................... 13  
  Secure your Web Services .......................................................................................................... 13  
  Security Updates ......................................................................................................................... 13  
  Leveraging The Event Driven Architecture ................................................................................. 13  
  Right Usage of Integration Interfaces .......................................................................................... 13  

*Note: Please send in suggestions / feedback to the author of the document*
Introduction

The Oracle E-Business Suite Adapter Methodology Guide is primarily written to provide the knowledge and information on getting started with the integration projects involving Oracle E-Business Suite in a heterogeneous ecosystem of IT applications.


This guide assumes that the reader of this document has a working knowledge of the following:

- Oracle E-Business Suite.
- Oracle SOA Suite components like Oracle BPEL PM & Oracle ESB
- Oracle Adapters
- Oracle E-Business Suite Adapter
- Oracle JDeveloper
- Oracle E-Business Suite Integration Interfaces
- Oracle Integration Repository
- Web Services
Overview
Let us try to understand some basic facts on integration, adapters and middleware.

Why Integration
Businesses have a heterogeneous ecosystem of IT applications. It is a pasta of legacy applications, enterprise applications and third party packaged applications on various flavors of technology platforms. However to run well orchestrated businesses, one requires complex business processes to run as an integrated ecosystem. To achieve this, seamless integration is required between business partners, business applications, business processes, and end-users.

Why Adapters?
Most IT applications in the ecosystem an enterprise do not yet have native support for Web Services, that means that the applications do not expose standard Web Services for integration purposes. Each application has proprietary and specific customized solutions to integrate backend applications. Most of the customized solutions are not metadata-based and hence changes to the underlying backend applications or integrations require huge amount of effort and are difficult to maintain. The time taken to deploy an integration solution is large and cost is quite high. An Adapter addresses these challenges and provides real-time, bi-directional, and comprehensive connectivity to various backend applications.

The Adapters are usually metadata-driven in a declarative format and integrate with one or more integration interfaces. They translate the data from proprietary data format to a standard data format like XML. Adapters in general themselves do not perform the tasks like transformation, process orchestration or routing but rely on other function-specific components of the middleware.

Oracle Fusion Middleware is a complete offering of solutions for building a standards-based, enterprise integration infrastructure for heterogeneous IT applications communication based on the principles of Service-Oriented Architecture (SOA).

Oracle E-Business Suite Adapter
The Oracle E-Business Suite Adapter provides comprehensive, bi-directional, multi-modal, synchronous and asynchronous connectivity to the Oracle E-Business Suite. Oracle E-Business Suite Adapter from the Oracle Fusion Middleware stack plays the role of service provider for Oracle E-Business Suite in heterogeneous ecosystem of your extended enterprise to provide the seamless integration.

Oracle E-Business Suite Adapter is based on a set of standards such as J2EE Connector Architecture (J2CA), Extensible Markup Language (XML), Web Service Invocation Framework (WSIF), Web Service Inspection Language (WSIL), and Web Service Definition Language (WSDL). The support for standards facilitates interoperability and eliminates vendor lock-in.

Oracle E-Business Suite Adapter supports the SOA principles of loose coupling of services, re-usability and flexibility to coexist with the best-of-breeds.
Oracle E-Business Suite Adapter Integration Methodology

The Oracle E-Business Suite Adapter Integration Methodology has a step-up process of 4 phases.

- Define
- Design
- Develop
- Deploy

Each phase is characterized by the set of activities that step up the integration project towards achieving the stated objectives and defined goals. The Oracle E-Business Suite Adapter Integration Methodology explains the project phases in relation to the Oracle E-Business Suite Adapter; however, the integration project has many other components involved, which the integration solutions team will have to consider in addition to this guide.
DEFINE Phase

The DEFINE phase of the integration project essentially consists of identifying scope definition, establishing the objectives, setting up goals, prototyping and identifying the requirements. These activities can be categorized into 2 sub-phases namely S-O-G and Prototyping.

Scope – Objectives – Goals

The first step in the S-O-G sub-phase is to completely detail out the business problem before treading on to the solution. Enlist all the business use-cases, all the business conditions and constraints, business processes and sub-processes participating in the integration flow.

For example, “Order to Cash” is one of the most common business processes that any company requires and there are 3 sub-processes namely Order Management, Order Fulfillment and Invoicing.

The second step in the S-O-G is to identify the scope of applications in the integration flow that requires connectivity between the broad-range of systems. These could include technology components, messaging middleware, legacy applications, mainframe applications, B2B applications, and third party packaged applications, EIS Applications and version & modules of Oracle E-Business Suite.

![Diagram of integration flow with Order Management, Invoicing, Order Details, Inventory Details, Shipping Details, Order Fulfillment, Packaged Apps, Legacy Apps, Oracle E-Business Suite]

For example, Order Management is the sub-process executed in Packaged Applications, Order Fulfillment happens in the Legacy Applications and the Invoicing is processed in Oracle E-Business Suite Release 11/10 or R12.

Prototyping The Possibilities

Prototyping is an iterative process to establish an early feasibility of achieving a solution by trying out the various possibilities. This sub-phase can also be termed as “proof of concept” (POC). In integration projects, building the complete design can often be expensive and time-consuming, hence rather than building the full design, figuring out what the possibilities are, through a POC is more useful. However before embarking on the POC, do glance through this entire document, as the prototyping is more or less representative of all the phases of an entire life cycle of integration project.

The prototyping phase is characterized by identification of the integration end-points in each application. The proof of concept can establish the viability of using Oracle E-Business Suite
Adapter for inbound and outbound transactions leveraging the other components of Oracle Fusion Middleware, like Oracle BPEL PM, Oracle ESB, and Oracle Technology Adapters, etc.

In above running example, the prototyping will help in proving the viability of Order to Cash business process with Order Management in Packaged Applications, Order Fulfillment in the Legacy Applications and the Invoicing in Oracle E-Business Suite Release 11/10 or R12 as the integrated flow with various possibilities.

**Checkpoints at DEFINE phase**

1. The business process and sub-processes participating in the Integration flow.
2. The number of instances of Oracle E-Business Suite involved.
DESIGN Phase
The DESIGN phase of the integration project primarily consists of determining the type of integration and the design principles to be implemented in the solution.

Integrations can be broadly classified into the following categories:

1. Synchronous: Synchronous Integration is a request-response mode of operation between two systems. In short, the host application expects a response from the provider when a method on the latter is invoked in the same thread.

2. Asynchronous: Asynchronous Integration solutions are helpful when EIS Applications take an indeterminate amount of time to respond to a host application. In this case, the host application sends the request and proceeds with its own actions without waiting for the response to process in the same thread. The provider application picks up the request at a later stage and processes the message.

3. Batch: Batch integrations allow import or export of large batches of data that are collected over a period of time into the system. For example, consolidated transactions update at scheduled intervals employ this technique.

4. Process Oriented: Process Orchestration takes application integration to the next level by using an application independent state engine to manage business processes that span multiple applications. Process Orchestration can involve system-to-system integration, system to human task integration, or human task to human task integration.

Determination of integration type for the solution helps in identifying the artefacts of design and architecting the solution.

Identifying The End Points
It is most imperative to identify the integration interface of Oracle E-Business Suite out of the following types of interfaces:

- PL/SQL APIs
- Business Events
- Open Interface Tables
- Concurrent Programs
- Oracle XML Gateway Interface
- Oracle e-Commerce Gateway Interface
- Interface Views

Once the integration interfaces are identified, the developer needs to identify the WSDL, XML schema, data files, object mappings and parameter values etc.

Oracle Integration Repository provides the catalog of credible public integration interfaces for Oracle E-Business Suite 11/10 or R12. Oracle E-Business Suite Adapter leverages Oracle Integration Repository to enlist the public integration interfaces via an Oracle Integration Repository xml meta-data file by product family view for Oracle E-Business Suite 11.5.10. However, it enlists all the interfaces for the R12.0 via either a live connection or cached xml meta-data file.
Identifying The Direction Of Transactions

Identify the inbound and outbound transactions for integration, which will provide the finer details for the technical design and architecture. The direction of transaction will have impact on the architecture of the solution, input and output parameters and actual interfaces to be used for the solution.

For example, for inbound Sales Order transaction, you will import data into Sales Order Headers Interface (OE_HEADERS_IFACE_ALL) and Sales Order Lines Interface (OE_LINES_IFACE_ALL) open interface tables followed by the execution of the ORDER IMPORT (OEOIMP) concurrent program to validate and update Oracle E-Business Suite base tables.

Similarly for outbound transactions, you may extract, transform, and write application data to xml data files from appropriate integration interfaces.

Architecting The Solution

Solution Architecture refers to the way in which the components of Oracle Fusion Middleware and IT applications deployed in the implementing organization are organized and integrated. Solution Architecture requires implementing the right choice of technology, packaged-applications and Legacy adapters in addition to Oracle E-Business Suite Adapter.

The technology adapters integrate Oracle Application Server with transport protocols, data stores, and messaging middleware. These adapters include OracleAS Adapter for FTP, OracleAS Adapter for JMS, OracleAS Adapter for Database, OracleAS Adapter for Advanced Queuing, and OracleAS Adapter for Files. Technology adapters are currently available with the Oracle SOA Suite installation.

The Oracle E-Business Suite Adapter exposes the Oracle E-Business Suite’s interfaces as Web services. The Web Services generated by Oracle E-Business Suite Adapter can be deployed to leverage the Oracle SOA Suite components depending upon the requirement of the integration project and the solution options exercised. The integration services can be orchestrated using Oracle BPEL PM in a cross applications business process, which can participate in event driven architecture via Oracle ESB, deliver real-time dashboards via Oracle BAM and/or leverage Oracle Integration B2B for business-to-business transactions.

Finally, solution architecture should implement an service-oriented architecture, which facilitates loose coupling, flexibility, and extensibility while also considering the approaches for error handling, rollback, monitoring, transaction support, logging and auditing. The deliverable of the design phase should be Technical Design Document.

Checkpoints at DESIGN phase

1. The inbound and outbound integration transaction involved.
3. Complete details of parameter values, mappings, schemas, and relationships.
5. Complete details of the interaction with the other technology, legacy and application adapters.
6. Complete details of the interaction with the Oracle SOA Suite components like Oracle BPEL PM, Oracle ESB, Oracle BAM, Oracle B2B, Oracle OWSM, Oracle EM.
DEVELOP Phase
The main activities in the DEVELOP phase is to actualize the Technical Design Document.

Develop The Solution
Oracle JDeveloper is a complete IDE for Service-Oriented Architecture (SOA). Oracle JDeveloper provides full support for developing new Web services and consuming existing Web services. Oracle JDeveloper provides a development environment for this core SOA based development with support for BPEL Designer and ESB Designer.

Based on the choice of solution architecture, the integration solution can be modeled either using the Oracle BPEL Designer / the Oracle ESB Designer and / or both by configuring Oracle E-Business Suite Adapter services appropriately along with the other services.

The BPEL Designer - a graphical drag & drop environment to design BPEL-based process flows and Web Services orchestration. Oracle BPEL Process Manager provides a comprehensive, standards-based and easy to use solution for creating, deploying, and managing cross-application business processes with both automated and human workflow steps – all in a service-oriented architecture.

The ESB Designer - a graphical drag & drop environment to design message flows or Web Services. Oracle ESB provides multi-protocol messaging, routing, and transformation capabilities to distribute event messages and integrate services. Oracle ESB provides connectivity leveraging Oracle Adapters, which provide standards based access to virtually any data source.

Once the design is completed, the integration project is packaged as the BPEL suitcase or ESB suitcase with all the artefacts created during the integration project solution development.

Test-Run the solution
It is critical to make sure that a Test Plan is in place for integration solution developed. The test plan should take into consideration all the business use cases while creating their test use cases. The test plan should include all functional requirements as well as non-functional requirements for the business use cases.

The testing should test run the integration solution under following conditions:

1. Execute all the end-to-end integration scenarios fulfilling the business requirement and all its conditions and constraints.
2. The test run should be simulated under actual production loads in the test environment.
3. Test run should ascertain the appropriateness and completeness of the integration solutions including the right choice of the Oracle E-Business Suite interfaces.
4. In the test run, testing should be done on the test target and test source Enterprise Applications as separate from the design time.
5. Test for performance, scalability and Load-balancing.
6. Also execute negative test cases.

Once the test run has passed successfully in a simulated near-production environment meeting all the set criteria and quality standards through quality assurance / user acceptance testing, the integration solution is ready for deployment.
Checkpoints at DEVELOP phase

1. List of all the Oracle E-Business Suite services created.
2. WSDL definition of all the services.
3. Connection information for all the Oracle E-Business Suite instances for all the services.
4. All the run-time setup data is configured in Oracle E-Business Suite instances.
5. All the run-time Agent Listeners are up and running.
6. All the run-time artefacts are applied on the Oracle E-Business Suite instances.
7. Test Run for all the Oracle E-Business Suite services is working fine without errors.
Deploy Phase

Deploy the solution

Some generic points to be taken care of while deploying:

1. The connection to the Oracle E-Business Suite must be configured appropriately.
2. The run-time parameters, system settings and application setup should be properly configured for deployment for the integrations.
3. Run-time artefacts for PL/SQL, Business Event, and Concurrent Programs should be applied to the run-time instances.
4. Agent Listeners should be up and running in the Oracle E-Business Suite.

The key users and end-user training must be completed for the implementing organization before the deployment of the integration solution on production environment.

Checkpoints at DEPLOY phase

1. Connection information for all the Oracle E-Business Suite instances for all the services is configured as per the production instances.
2. All the run-time setup data is configured on the production instances of Oracle E-Business Suite.
3. All the run-time Agent Listeners are up and running on the production instances of Oracle E-Business Suite.
4. All the run-time artefacts are applied on the production instances of Oracle E-Business Suite.
5. All necessary Administrator, Developer and End-User trainings are completed.
6. Necessary recovery and rollback mechanisms are in place to meet any contingency requirements.


For more details on the components of Oracle Fusion Middleware and Oracle E-Business Suite, please refer to the respective User Guides and other product collateral documents for the same.
Best Practices And Recommendations

Securing Integration
Security is the most critical feature to guard application content from unauthorized access and you must ensure that the security levels are enabled to ensure secured environment.

Enable Function Security
By leveraging Oracle User Management function security, Oracle E-Business Suite Adapter provides a security feature, which allows only those users who have authorized privileges to execute APIs that are exposed through the BPEL PM process to update Oracle E-Business Suite. This protects application-programming interfaces (APIs) from unauthorized access or execution. To enable the Function Security feature, you need to set EBS_ADAPTER_FUNCTION_SEC_ENABLED profile option to Yes.

Secured connection between Oracle E-Business Suite and Fusion Middleware
Oracle EBS Adapter works in a secured and trusted connection environment between Oracle FMW and Oracle EBS that uses FND user name and password only and thus requiring no other shared credentials. More details can be found on Metalink Note # 464164.1

Secure your Web Services
Web Services require security policy management and monitoring. Consider securing your web services as leaving them unsecured can be a risk.

Security Updates
And last but certainly not the least, it is recommended to have all security patches for the application, middleware and OS installed on the production system.

Leveraging The Event Driven Architecture
Writing triggers on a table is not recommended rather use business events from Oracle E-Business Suite to create an event driven architecture for non-intrusive integration. Business events based integration is a standard way and it ensures easy customization and has a lower maintenance cost.

Right Usage of Integration Interfaces
Some recommendation for the type of integration interfaces usage for integrating Oracle E-Business Suite though these may not necessarily mean the only solution.

- You may consider Open Interface Tables as the integration interfaces if you are looking at the batch integration.
- Whenever you have simple data level requirements, use public PL/SQL APIs, as the Oracle E-Business Suite as well as E-Business Suite Adapter extensively support these.
- You may also consider going for standard and XML messages based integration. The XML messages indirectly calls either Open Interface Tables or API's but for the developer / end-user it is abstracted.