Building E-Business Suite Interfaces using BPEL

Asif Hussain
Innowave Technology

Agenda

- About Innowave
- Why Use BPEL?
- Synchronous Vs Asynchronous
- BPEL Adapters
- Process Activities
- Building EBS Interfaces
- Oracle Application Adapter
- How to make better processes
- More Concepts

About Innowave

INNOWAVE TECHNOLOGY

- Technology Consulting Firm founded in 2005
- Provide unique, innovative solutions using Oracle Applications and Technology
- Integration Experts: Fusion Middleware, SOA Suite, BPEL & Web services
- Implement, upgrade, and enhance Oracle E-Business Suite, JD Edwards, PeopleSoft, and Siebel Applications and related technologies
- Extensive experience and expertise managing global and local implementations across various industry verticals.
- Provide clients with proven methodologies, tools, and templates specifically tailored to their requirements.

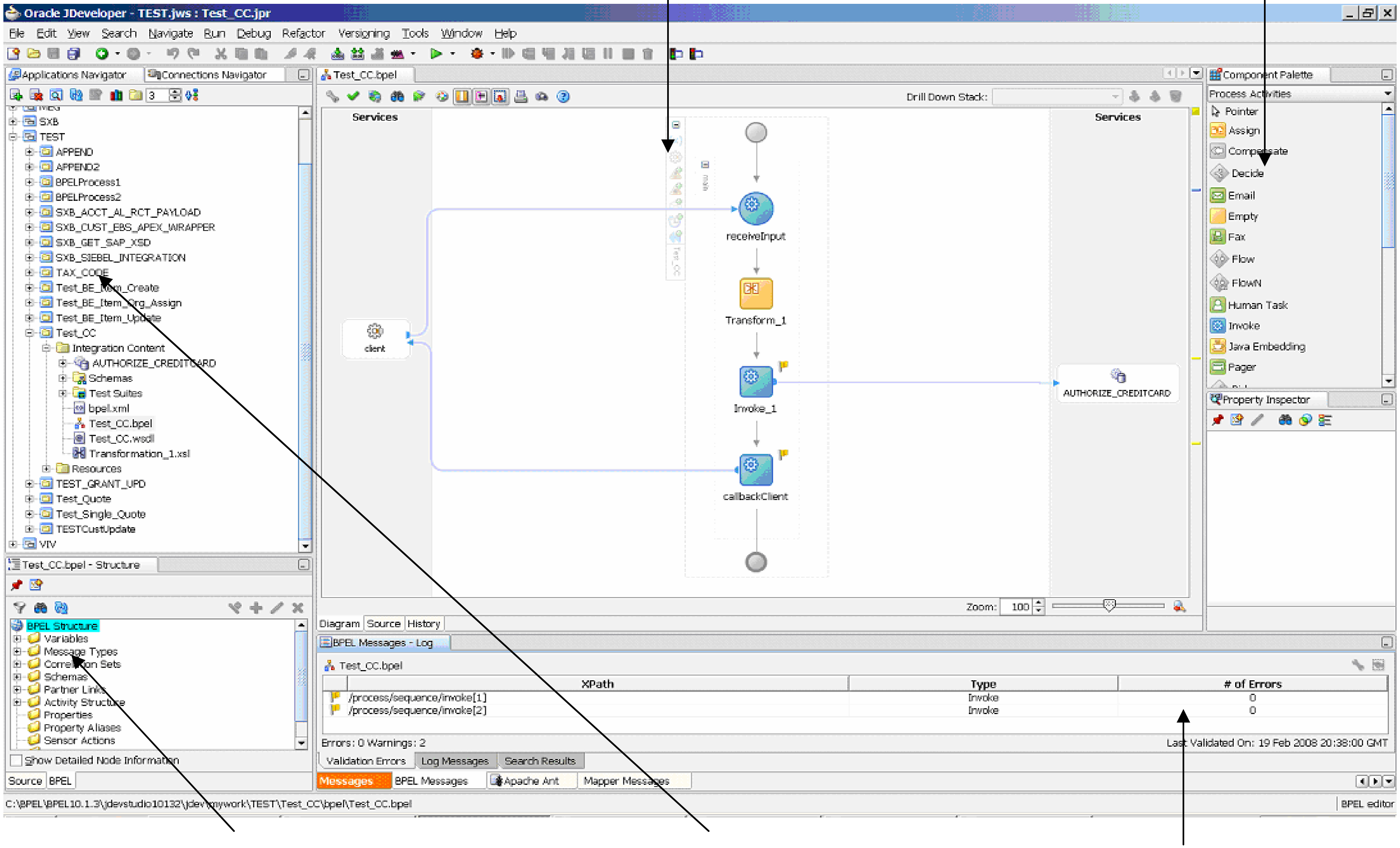
Why Use BPEL?

- Development is much faster as compared to the conventional approaches.
- The tool uses a drag and drop approach, which makes is easy to use.
- The processes are easy to modify in case of a business process change.
- It has out of the box management and monitoring tools.
- Delivers a standards based, platform-neutral solution.
- Adds a layer of security and exception management.
- In addition to drag and drop, experienced developers can write their own code.

BPEL Designer

Diagram View

Component Palette



Test_CC.bpel

XPath	Type	# of Errors
/process/sequence/invoke[1]	Invoke	0
/process/sequence/invoke[2]	Invoke	0

Errors: 0 Warnings: 2

Last Validated On: 19 Feb 2008 20:38:00 GMT

C:\BPEL\BPEL10.1.3\jdevstudio10132\jdev\mywork\TEST\Test_CC\bpel\Test_CC.bpel

Structure Pane

Application Navigator

Message Log

Synchronous Vs Asynchronous – Which one to chose?

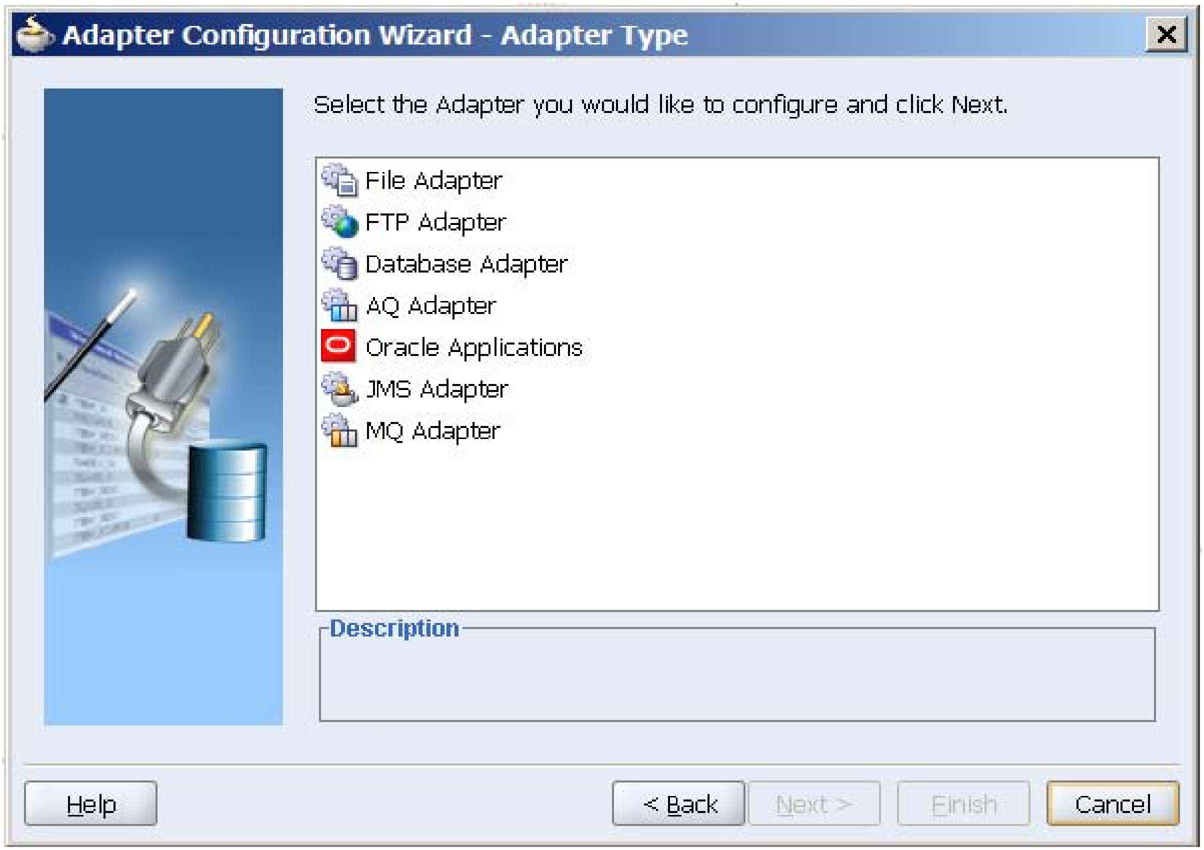
Synchronous Process

- A synchronous service waits for a reply from a partner service before continuing.
- In synchronous services, the **Invoke** activity is used for both sending the request and receiving the reply.
- Should only be used when an immediate response is expected.

Asynchronous Process

- Unlike a Synchronous process, the invocation of an asynchronous process does not need immediate response.
- For asynchronous processes, the **Invoke** activity is used for sending a request, and the **Receive** activity is used for receiving the reply from the service.
- After a BPEL process invokes an asynchronous process, it continues working through its process flow, until it encounters a **Receive** activity.
- Which is better?
- Empty process.
- How to build Template based processes.

BPEL Adapters



Adapter Configuration Wizard - Adapter Type

Select the Adapter you would like to configure and click Next.

- File Adapter
- FTP Adapter
- Database Adapter
- AQ Adapter
- Oracle Applications
- JMS Adapter
- MQ Adapter

Description

BPEL Adapters

- Database Adapter
  - Call a stored procedure or function.
  - Can perform a SELECT/ INSERT/ UPDATE or DELETE operation on a table.
  - Poll for new or changed records in a table.
  - Execute Custom SQL.
- File Adapter
- FTP Adapter
- AQ Adapter
- Oracle Applications
- JMS Adapter
- MQ Adapter

Process Activities

- What are process activities?
- How does the following process activities work?
  - Partnerlink
  - Invoke
  - Assign
  - Transformation
  - Scope

Partnerlink

- A partnerlink is your access to the outside world.
- A partnerlink type characterizes the conversational relationship between two services by defining the roles played by each service in the conversation.
- Can be used to
  - Read and Write files.
  - Read tables and Write into tables.
  - Invoke other BPEL processes deployed on the same server.
  - Invoke outside processes.

Invoke

- An **invoke** activity invokes a synchronous service or initiates an asynchronous Web service.
- It can be used to automatically create variables.
- It opens a port in the process to send and receive data.
- It uses this port to submit required data and receive a response.

Assign

- This activity provides a method for data manipulation, such as assigning the contents of one variable to another.
- You can assign constants, use functions, or write SQL statements.
- There is no limit to the number of assignments.

Scope

- A scope activity consists of a collection of nested activities.
- These activities can have their own local variables, fault handlers, compensation handlers, and so on.
- A Catch or Catch All branch can be associated with the scope activity for exception handling purposes.

Transformation

- This activity is used to create a transformation that maps source elements to target elements.
- There are a lot of functions that can be used for linking the source to the target elements.
- **Select** statements can be written with parameters.
- A lot of string and date functions available.
- **Assign** Vs **Transformation**.

Building an EBS Interface using interface tables

- Item Creation process.
- In BPEL, it will be a three step process.
  - Data will be read from a file.
  - Data will be written into MTL_SYSTEM_ITEMS table.
  - Submitting item import process.

DEMO!

Submitting a concurrent process to kick off Item Import

- FND_REQUEST.SUBMIT_REQUEST will be used to submit a concurrent process in EBS.
- This API can be submitted using the Oracle Applications adapter.
- This API can also be submitted using the database adapter.
- If called through the database adapter, a custom procedure is needed that calls the Submit Request API.

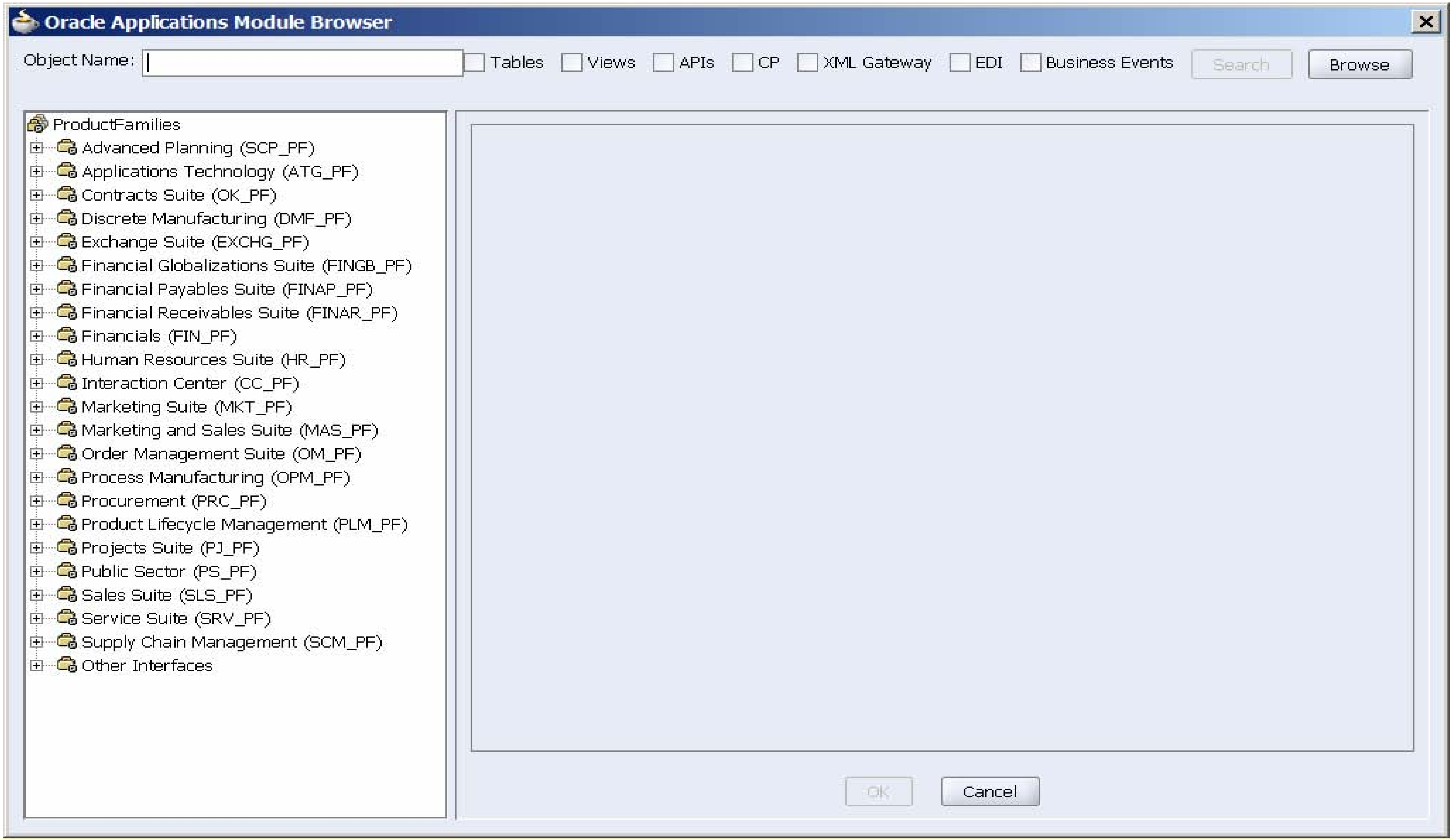
Building an EBS Interface using an API

- Instead of interface tables, Oracle seeded API’s can be used to create transactions within EBS.
- For example, EGO_ITEM_PUB.Process_Items and EGO_ITEM_PUB. Assign_Item_To_Org API’s can be used for creating items and assigning orgs.
- These API’s can be called through a custom wrapper package through the database adapter.
- If the API has any record type or table type parameters, BPEL builds its own wrapper around it.
- Available in BPEL PM Version 10.1.3 onwards.

Oracle Applications Adapter

- Oracle Applications adapter acts as a highly flexible integration interface for Oracle Applications.
- Supports all modules from versions 11.5.1 to11.5.10.
- Provides real-time and bidirectional connectivity to Oracle Applications.
- Supports the following interface types for integration.
  - Interface Tables
  - Views
  - API’s
  - Concurrent Programs
  - XML Gateway
  - EDI
  - Business Events

Oracle Applications Adapter



ProductFamilies
- Advanced Planning (SCP_PF)
- Applications Technology (ATG_PF)
- Contracts Suite (OK_PF)
- Discrete Manufacturing (DMF_PF)
- Exchange Suite (EXCHG_PF)
- Financial Globalizations Suite (FINGB_PF)
- Financial Payables Suite (FINAP_PF)
- Financial Receivables Suite (FINAR_PF)
- Financials (FIN_PF)
- Human Resources Suite (HR_PF)
- Interaction Center (CC_PF)
- Marketing Suite (MKT_PF)
- Marketing and Sales Suite (MAS_PF)
- Order Management Suite (OM_PF)
- Process Manufacturing (OPM_PF)
- Procurement (PRC_PF)
- Product Lifecycle Management (PLM_PF)
- Projects Suite (PJ_PF)
- Public Sector (PS_PF)
- Sales Suite (SLS_PF)
- Service Suite (SRV_PF)
- Supply Chain Management (SCM_PF)
- Other Interfaces

Oracle Applications Adapter Interface Types

- Interface Tables - Enables you to insert or update data into Oracle Applications.
- Views - Helps retrieving data from Oracle Applications using the application tables.
- PL/SQL API’s - Helps inserting and updating data in Oracle Applications using PL/SQL.
- Concurrent Programs – Oracle Applications Adapter can be configured to use the concurrent program integration interface to perform specific tasks. Enables you to move data from interface tables into base tables.

Electronic Data Interchange (EDI)

- Oracle e-Commerce Gateway provides a common, standards-based approach for Electronic Data Interchange (EDI) integration between Oracle Applications and third party applications.
- Oracle Applications adapter can be configured to use the Oracle e-Commerce Gateway to dynamically generate outbound or consume inbound flat files.
- These files will be based on user-defined trading partner, mapping, transformation, and data validation rules.

XML Gateway

- Oracle XML gateway is a set of services that allow easy and bidirectional integration with Oracle Applications to support XML messaging.
- Oracle Application adapter can be configured to use the XML gateway integration interface to interact for inbound or outbound transactions with third-party applications.
- Integrates with interface tables, Oracle Workflow Business Event System (BES), and interface views to insert and retrieve data from Oracle Applications.
- Maps the underlying table data to XML and back.

Business Events

- Business Events are Oracle EBS functionality, which are fired whenever a transaction happens within EBS.
- It is an application service which uses the Oracle advanced queuing infrastructure.
- Business Event mechanism can be combined with BPEL to communicate with other systems.
- Business Events can be synchronized with BPEL processes for inbound and outbound transactions.
- One example is to create an item in a third party system, whenever an item is created in Oracle.

Advanced Queuing

- A queue is a database object, which contains specific information about the events stored in it.
- Each event type has a unique value called a correlation id.
- Using AQ Adapter, a BPEL process having the same correlation id is initiated when a queue is populated.
- A queue can be populated through a database procedure. Hence a concurrent process can be created which populates a queue, and hence kicks off a BPEL process.
- Used when the same URL is used to post all the messages.

More Concepts

How to write efficient processes

- A few bells and whistles can be added to make a process robust.
- A few concepts to consider are
  - Exception Handling
  - Java Embedding
  - Notifications
  - Wait Activity
  - Calling a BPEL process from another BPEL process
  - Importing a user defined XML Schema
  - Source Editing
  - Log Files

Exception Handling

- A **CatchAll** branch can be added to a Scope activity.
- The CatchAll will catch any errors that happen during the Scope activity.
- CatchAll can be put around an entire process to catch any error during the process.
- Highlight an Error-Out process.

Java Embedding

- Custom Java Code can be embedded within the process using the **Java Embedding** process activity.
- Checkpoint() is a very useful java function that dehydrates the BPEL process at any given time. It is used to debug a process.

Notifications

- Email
  - Most Widely used.
  - Use Query Database function to generate email addresses.
  - Dynamically create email subject and body.
  - NS_EMAILS.xml file.
- Fax
- Pager
- SMS
- Voice

Wait Activity

- Used for introducing delays in the process.
- A typical use of this activity is to invoke an operation at a certain time.
- This activity allows you to wait for a given time period or until a certain time has passed.

Calling BPEL process from another BPEL Process

- The BPEL process to be called must be deployed on the server.
- Partnerlink activity is used to call a BPEL process.
- The child process will show up as a separate entry on the console.
- Click the Tree Finder button to find all the related processes.

Importing a User Defined XML Schema

- By default, a BPEL process accepts an input message and returns an output message.
- The input message can be modified by using an imported XML Schema document or XSD.
- XSD – XML Schema Definition file specifies how to describe the elements in the XML Document.

Source Editing

- Although BPEL uses building blocks to create the processes, sometimes direct source editing can make the process creation faster.
- Source code can be cut and pasted, which makes the building blocks to be created automatically.

Log Files

- Domain.log captures all the activity going on into the BPEL server.
- It is a great tool for debugging process flows.
- Logging level can be increased or decreased through the console.

INNOWAVE TECHNOLOGY

Q & A

Asif Hussain

asif.hussain@innowavetech.com