



ORACLE BPEL TRAINING

BASHEER KHAN

INNOWAVE TECHNOLOGY

This article introduces readers to concepts of Web services, SOA and how to use Oracle BPEL Process Manager for developing business processes as well as interfaces between applications. It illustrates the use of Oracle BPEL Designer for configuring various adapters, locating and registering web services as well as creating data mappings to interface data from one application to another. Readers will also learn how the Oracle BPEL Console can be used to monitor interfaces once they are created.

This article will help you:

- ✓ Understand the concept of Web Services & Service Oriented Architecture (SOA)
- ✓ Learn how to use BPEL Designer to orchestrate Web Services
- ✓ Understand how to integrate applications using BPEL Process Manager
- ✓ Learn how to use BPEL Console to monitor BPEL processes

Concepts

Web services

Web services expose business functions to authorized parties over the Internet or across intranets using a loose-coupled, self-describing and standards-based protocols that are language and platform agnostic. Web services evolved over the years as a natural progression of inter-party communications from person-to-person (e.g. email) to person-to-application (e.g. Web) to the current application-to-application interactions using Web services.

Web services are reusable “building blocks” (created & made available by Web service providers) that can be used to develop complex end-to-end processes to automate different business functions in an organization. Providers around the world are registering their Web services in electronic “yellow pages” or public registries (also referred to as Web Service Broker).

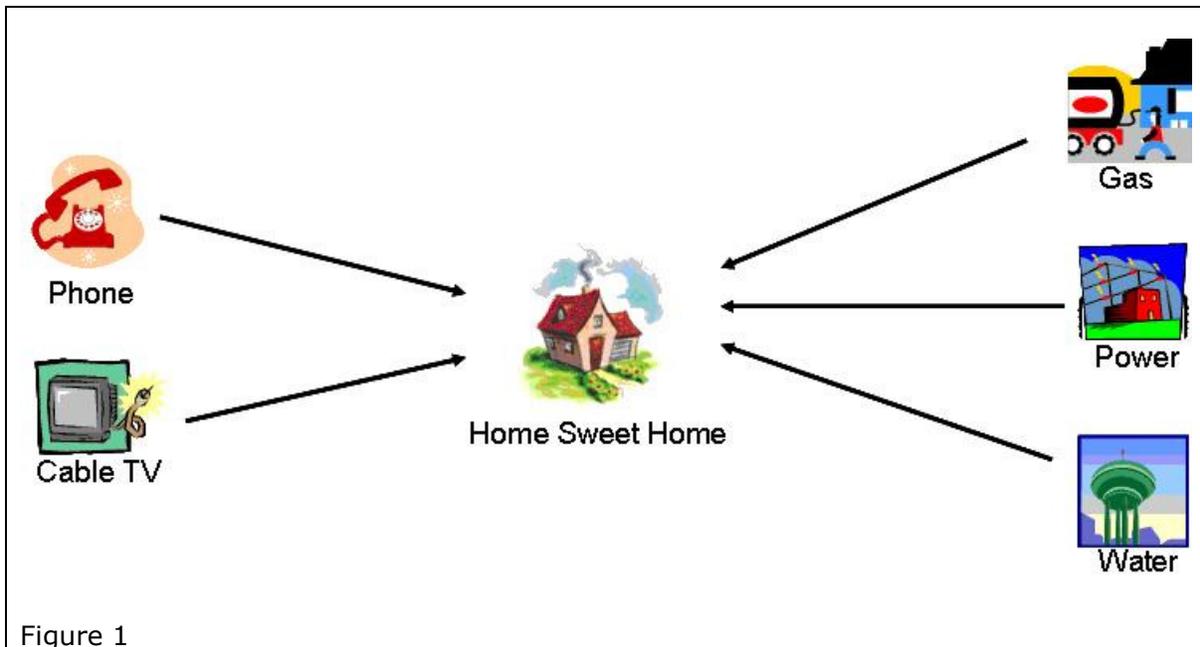
Making Web services work is a two step process. First, you develop and publish your services and then you compose or orchestrate them into business flows. Publishing a service means taking a function within an existing application or system and making it available in a standard way, while orchestration is composing multiple services into an end-to-end business process. The Web services standards, including WSDL (Web Services Description Language), XML (Extensible Markup Language) and SOAP (Simple Object Access Protocol), have emerged as an effecting and highly interoperable platform for publishing services.

Most major vendors are introducing Web services in their software applications and Oracle also has a registry that will be discussed later in this article (see section titled "Integration Repository"). Instead of building applications from ground up, organizations can now leverage these "building blocks" to develop standards-based applications or interfaces to address their business needs.

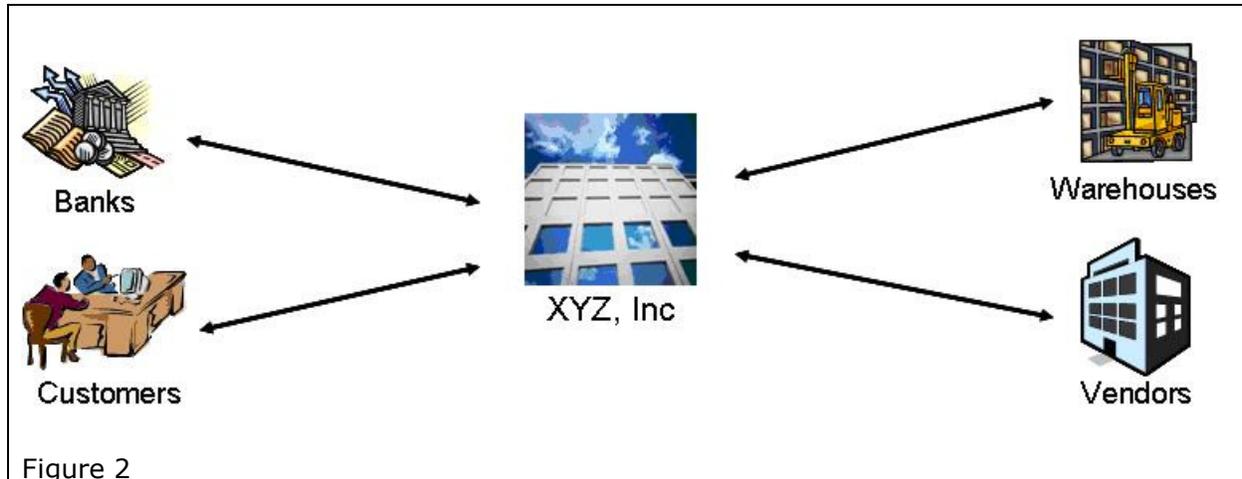
Service Oriented Architecture (SOA)

SOA is a software architectural concept that defines the use of services to support business requirements. In an SOA, resources are made available to other participants in the network as independent services that are accessed in a standardized way.

What most of us do not realize is that when we move into a new home, we establish and use (everyday) a Service-Oriented Architecture as shown in Figure 1:



Yet, most businesses today build their own bridges to interact between applications or with their trading partners as illustrated in Figure 2:



There is a paradigm shift in the industry as more and more organizations realize this anomaly. SOA is gaining rapid acceptance and adoption today. While most definitions of SOA identify the use of web services (using SOAP and WSDL) in its implementation, note that it is possible to implement SOA using any service-based technology.

An increasing number of companies are looking at SOA and Web services as an architectural blue-print and a set of standards for addressing the integration requirements involved in building connected applications. While SOA has been a best practice for over a decade, there has been confusion around which standards to adopt. SOA enables business agility by maximizing leverage of existing resources while minimizing the cost of deploying new services into a business process.

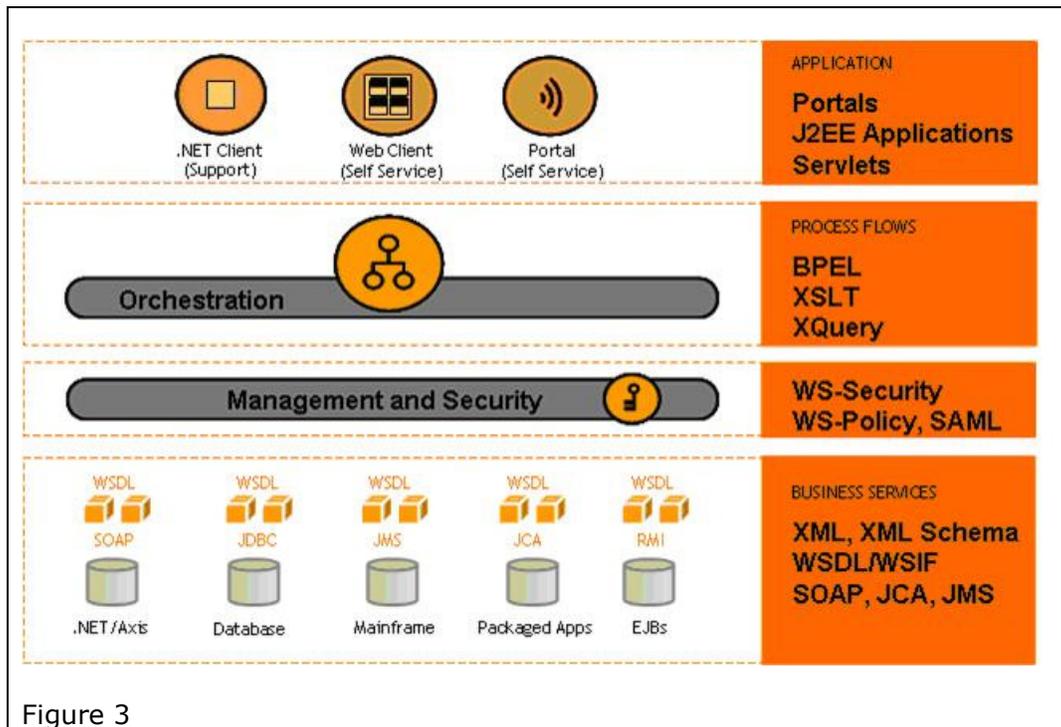


Figure 3

Most businesses use one or more of the applications/technologies depicted in Figure 3: .Net/Axis, Database, Mainframe, Packaged Applications or EJBs. Imagine each of these applications being able to communicate using Web services, that are secured, managed and orchestrated using standards-based platform to interact with different end-user applications (.Net Client, Web Client or Portal). This possible today using Oracle Technology and without the need to invest in rewriting existing applications. How this is possible will be discussed later in this article under the section titled "Adapters".

Business Process Execution Language (BPEL)

BPEL, considered the cornerstone of SOA, is a XML-based language for composing a set of discrete services into an end-to-end process flow (see Figure 4). It has built-in support for asynchronous interactions, flow control and compensating business transactions. It integrates with XPath, XSLT and XQuery for XML data manipulation.

BPEL evolved as a result of 10+ years of R&D by Microsoft and IBM and is a convergence of language features from IBM's Web Service Flow Language (WSFL) that uses a directed graph approach and Microsoft's XLANG, which is an orchestration language with a block-structured approach. Both WSFL and XLANG are now superseded by BPEL specification.

The key objective of BPEL is to standardize the format of business process flow definition to enable different organizations to work together seamlessly using Web services. Each BPEL process can be implemented as a Web service. These processes can be executed on any platform or product that complies with the BPEL specification. This approach significantly reduces the complexity and cost of integrating processes and applications.

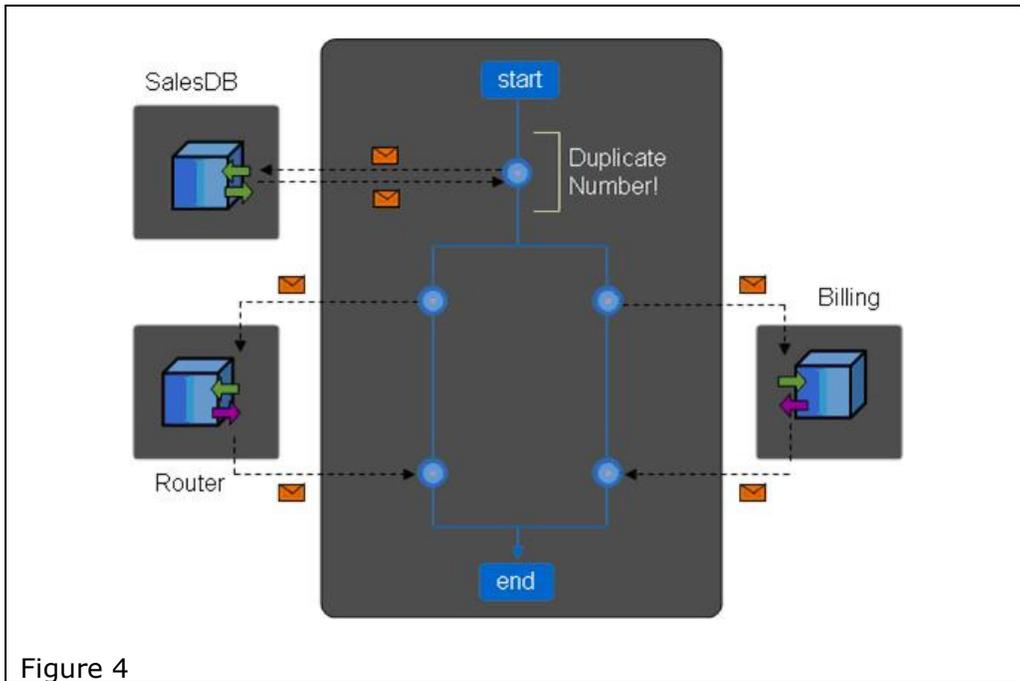


Figure 4

Architecture

BPEL Process Manager

Oracle BPEL Process Manager provides a developer-friendly and reliable solution for designing, deploying and managing BPEL business processes.

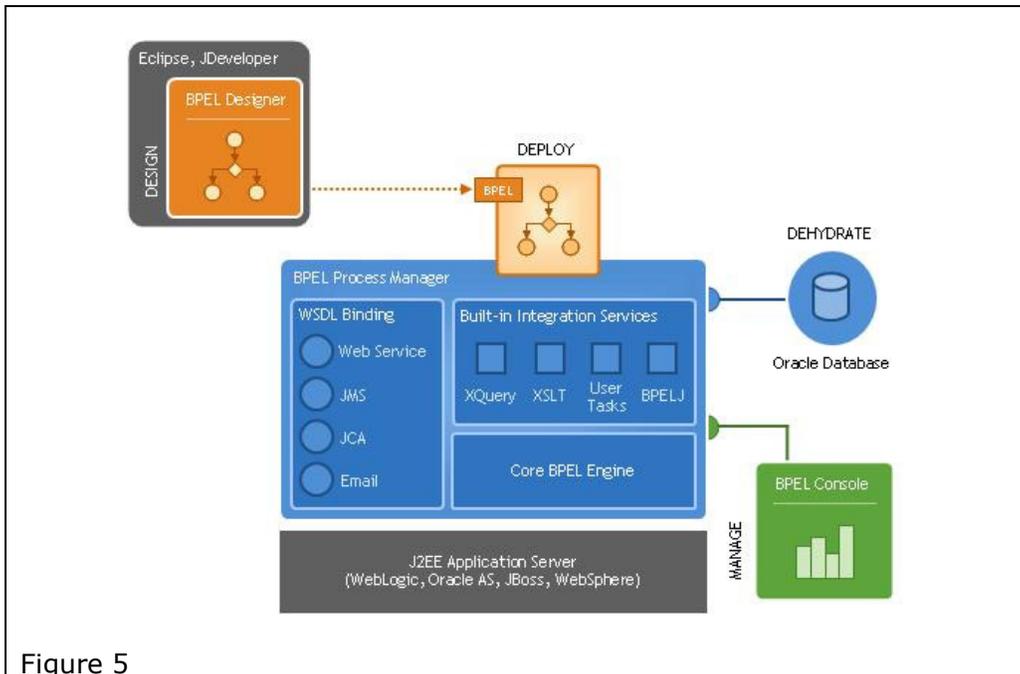


Figure 5

BPEL Designer

Oracle BPEL Designer provides a graphical and user-friendly way to build BPEL processes. What is unique about the Oracle BPEL Designer is that it uses BPEL as its native format. This means that processes build with the Designer are portable and in addition it enables developers to view and modify BPEL source at any time.

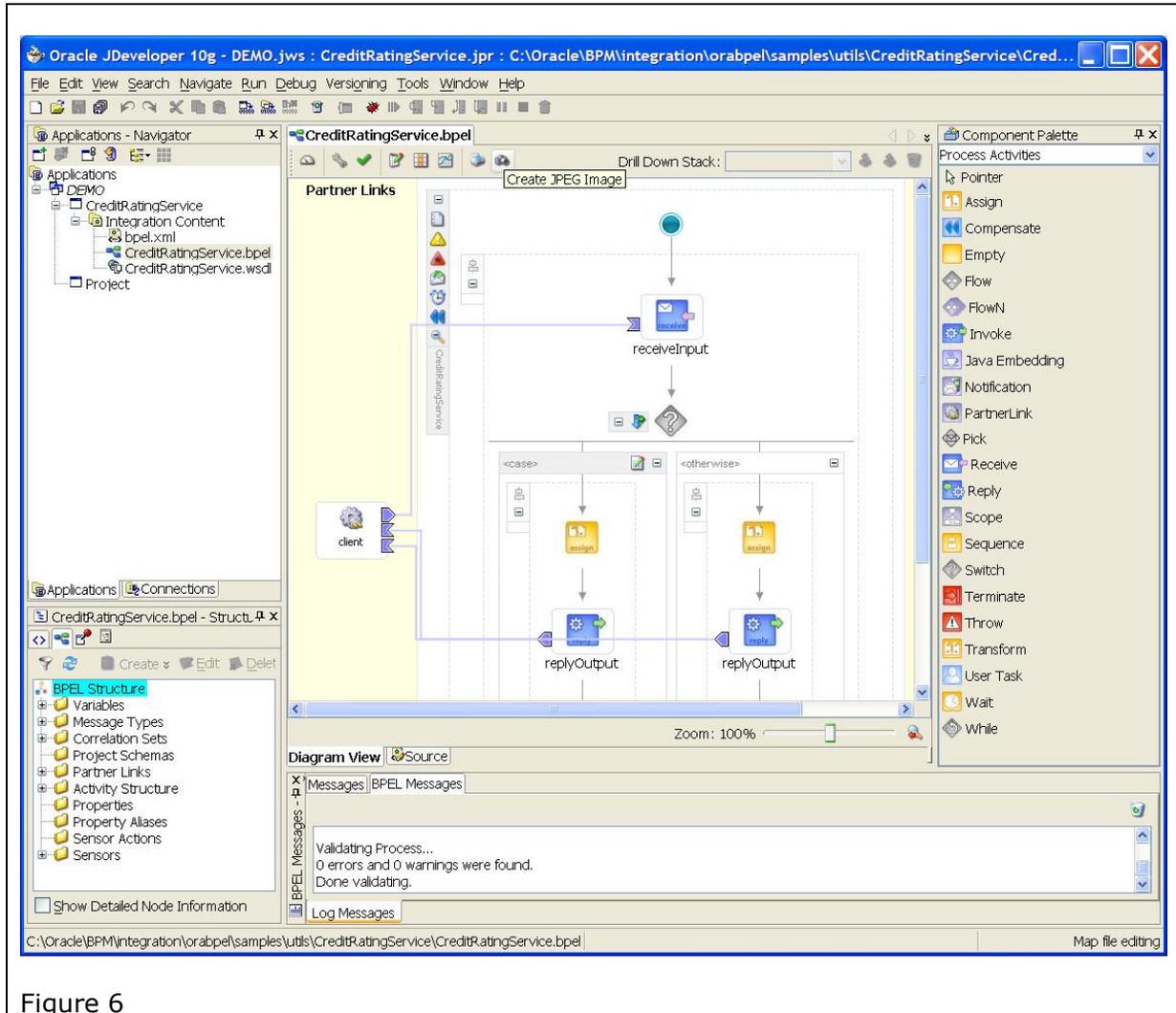


Figure 6

Adapters

BPEL Process Manager includes built-in integration services that enable developers to easily leverage advanced connectivity capabilities from standard BPEL processes. These capabilities include bindings to hundreds of legacy systems through JCA (J2EE Connector Architecture) adapters and native protocols.

It is by using these adapters that companies can jumpstart the introduction of SOA and Web services into their organization. For example, if an existing application (say Mainframe-based) is capable of exporting text files (most do), the File Adapter can be leveraged to

make that text file behave like a Web service, extracting and delivering data to external applications or processes in the form of an XML document. Similarly, any JDBC data source can be transformed into a Web service using the Database Adapter, exposing contents of a table or view as XML messages.

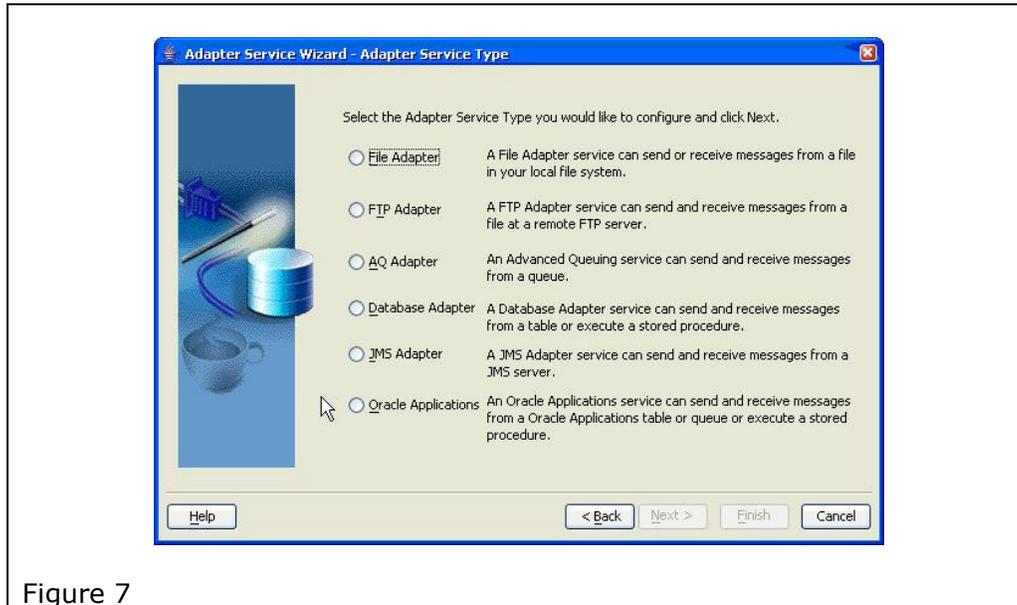


Figure 7

Using the built-in adapters, businesses avoid the need to invest in costly modifications to their existing legacy applications to introduce a layer of code that would expose data and business functions as Web services. In many instances, especially with 3rd party legacy applications, such a modification may not even be feasible or cost-effective.

Transformations

BPEL has built-in support for advanced transformation capabilities including support for XSLT and XQuery transformations. The BPEL Designer provides a user-friendly interface to map source and target attributes.

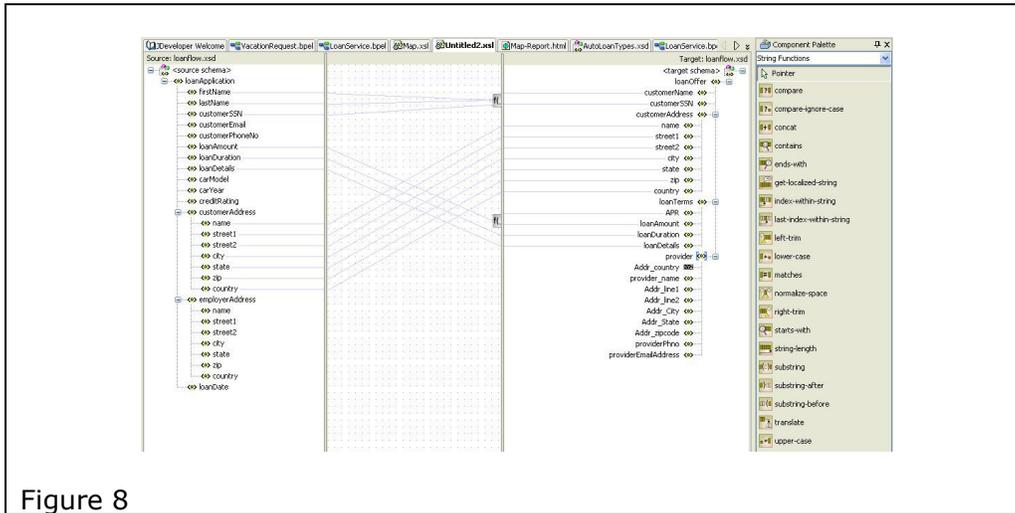


Figure 8

BPEL Console

The BPEL Console provides a mature web-based interface for management, administration and debugging of processes deployed to the BPEL server, using a “what you see is what you get” (WYSIWYG) approach. Audit trails and process history/reporting information is automatically maintained and available through the BPEL Console.

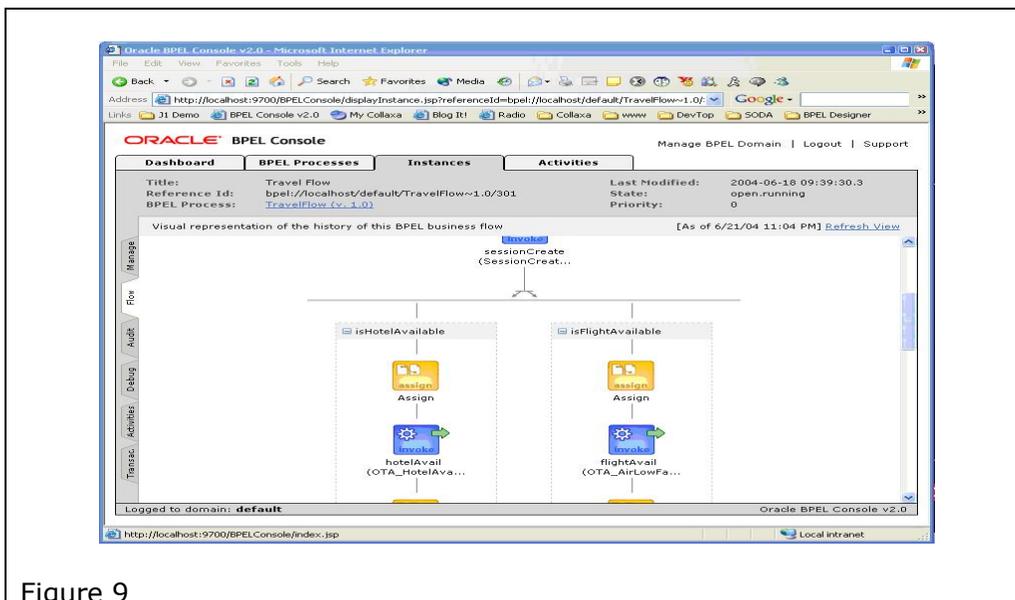
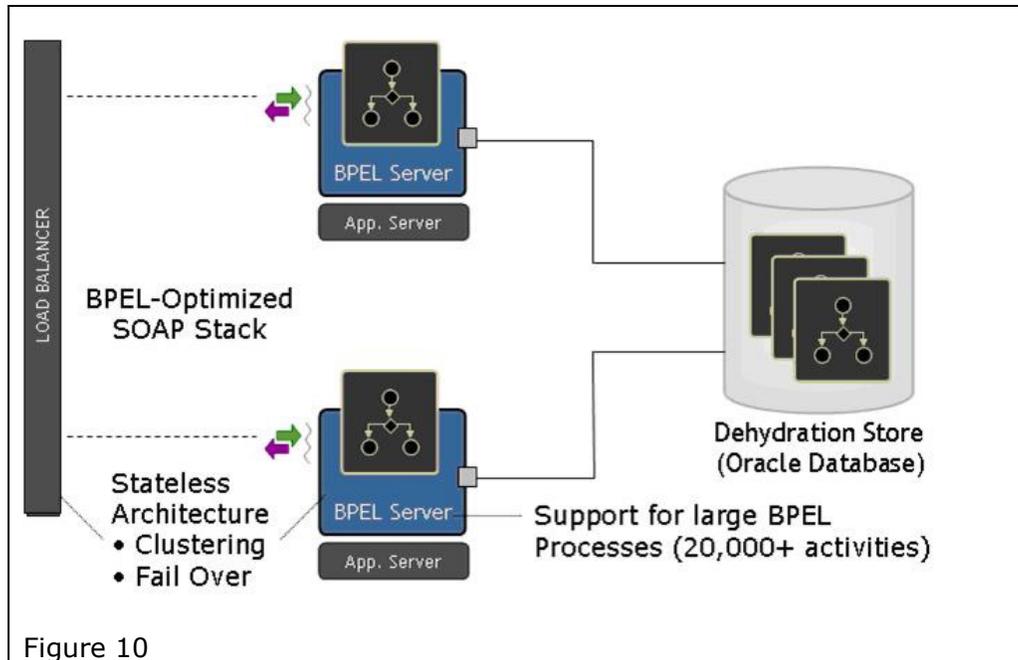


Figure 9

High Availability

The core BPEL engine provides a very scalable and robust implementation of the BPEL Server. It provides a “dehydration” capability so that the state of long-running process flows is automatically maintained in a database, enabling clustering for both high availability and scalability. The BPEL Server leverages an underlying J2EE application server, with support for most major commercial application servers.



Installation

These instructions are based on the 10.1.2 release of the Oracle BPEL Process Manager, and will guide you through the installation of the free developer edition.

Prerequisites

- Windows XP/2000 with SP3/2003 SP1 (or Sun SPARC Solaris version 8/9 or Red Hat Linux Enterprise AS/ES 2.1/3.0 or SUSE Linux Enterprise Server 8/9)
- 1 GB disk space, minimum 1 GB RAM and minimum 1535 MB of swap space
- Internet Explorer 6.0
- Monitor configured to display at least 256 colors

Download & Install

The Oracle BPEL Process Manager is available for download at <http://www.oracle.com/technology/bpel/>.

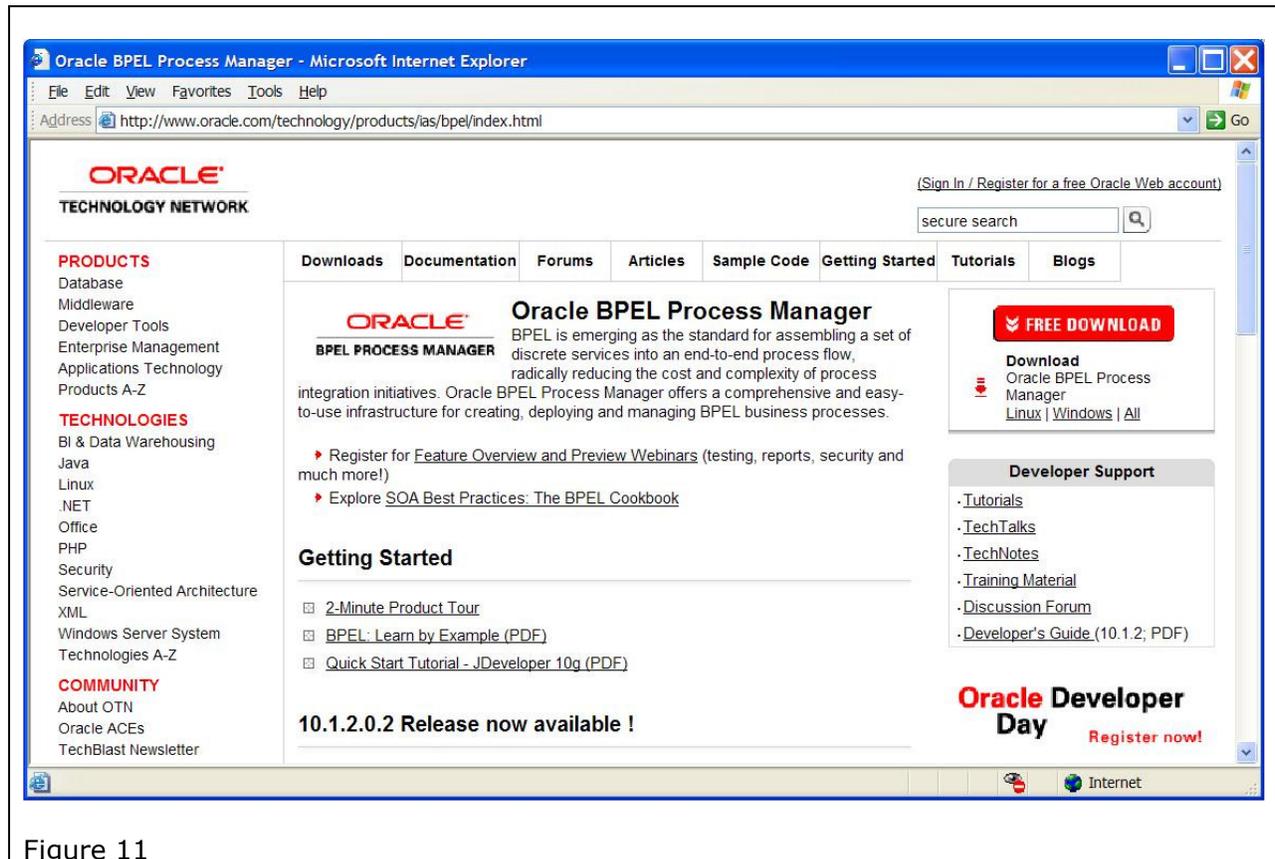


Figure 11

The Oracle BPEL Process Manager for Developers installation type installs both the JDeveloper BPEL Designer and the Oracle BPEL Process Manager. The installation also includes and deploys a stand-alone Oracle Containers for Java (OC4J) instance and an Oracle Lite Database.

1. Click on the "FREE DOWNLOAD" button to download BPEL Process Manager.
2. Extract the downloaded files to a local directory.
3. Start the Oracle Universal Installer from the bpel directory by clicking setup.exe
4. Click Next when the Welcome screen appears
5. Accept the default Name and Path by clicking Next (or specify a new Oracle home name and directory path). Do not use existing home or directory path. Do not change the directory path in the Source field.
6. Select BPEL Process Manager for Developers on the Select Install Type screen and click Next
7. Accept the default information (or enter proxy information, if applicable) and click Next



8. Click Next on the Summary screen and wait for the installation to complete.
9. Click Exit on the End of Installation screen and confirm when prompted
10. Apply additional patches if necessary

Starting BPEL Designer

Start the BPEL Designer from the Windows Start menu by clicking the following:

Start > Programs > Oracle – ORACLE_HOME > BPEL Process Manager 10.1.2 > JDeveloper
BPEL Designer

Starting BPEL Server

The Oracle BPEL Server needs to be started in order to deploy and test flows that you create with the Designer. Start the BPEL Server from the Windows Start menu by clicking the following:

Start > Programs > Oracle – ORACLE_HOME > BPEL Process Manager 10.1.2 > Start BPEL
PM Server

Stopping BPEL Server

The BPEL Server can be stopped by clicking the following from the Windows Start menu:

Start > Programs > Oracle – ORACLE_HOME > BPEL Process Manager 10.1.2 > Stop BPEL
PM Server

Learning BPEL

Getting Started

In order to assist you in your understanding of the concepts of Web services, SOA and how to use Oracle BPEL Process Manager for developing business processes, I have created four comprehensive tutorials, which can be downloaded as PDF files at

http://www.innowavetech.com/downloads/oracle_bpel_tutorials.zip

However, before you begin the tutorials, you should compile and deploy the credit rating service, which will be invoked as part of the tutorial 2, making it available on your local BPEL Server. You will use the command line rather than the BPEL Designer to compile and deploy this service. In general, all BPEL processes can be compiled either through the BPEL Designer or command-line Ant interfaces.

To compile and deploy the credit rating service:

1. Open up a developer command prompt if you do not already have one open
Start > Programs > Oracle – Oracle_Home > BPEL Process Manager 10.1.2 >
Developer Prompt.
2. Navigate to the Credit Rating Service Directory
> cd utils\CreditRatingService
3. Execute the obant command.



> obant

To test the credit rating service:

4. If you have not already done so, connect to the BPEL Console (at <http://localhost:9700/BPELConsole>) and log in. An alternative is to use the Windows Start menu and click:
Start > Programs > Oracle – Oracle_Home > BPEL Process Manager 10.1.2 > BPEL Console.
5. In the Name column (under the Dashboard tab of the console), click the link for the CreditRatingService BPEL process
6. In the test form that appears, enter any nine-digit number as the social security number (for example, 123456789) and click Post XML Message.

You should get back an integer credit rating (or a fault if the social security number you entered began with a 0). In either case, the service is confirmed to be installed successfully.

The tutorials are designed to give you a more in-depth experience of using Oracle BPEL Designer to build and deploy simple processes. The first tutorial teaches you how to build the classic "Hello World!" process. In the second tutorial, you will invoke the CreditRatingService process that you deployed as described above. The third tutorial teaches you how to create database connections. Finally, the fourth tutorial uses this database connection and teaches you how to load a comma-separated, text file into a database table.

Application Integration

Integration Repository

The Oracle Integration Repository is a catalog of every E-Business interface. The repository is now publicly available at <http://irep.oracle.com>. The types of interfaces listed in the Integration Repository include Web Services, Service Beans, PL/SQL Procedures, Java Classes, XML Gateway Messages, EDI Messages, Open Interfaces, Concurrent Programs, and Business Events. The documentation available in the Integration Repository includes information about Service Data Objects, XML Schemas, DTDs, Interface Tables and Views.

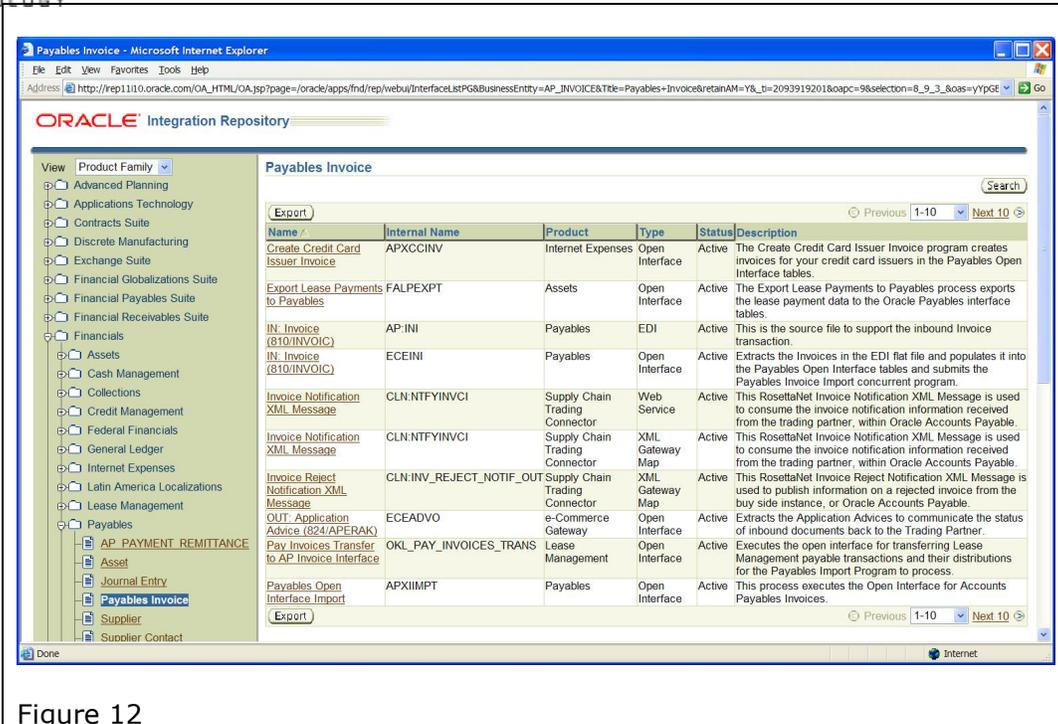


Figure 12

Figure 12 illustrates the various interfaces listed in the Integration Repository. In this figure, the interfaces are viewed by Product Family. The expanded node depicts the various modules under the Financials Suite Product Family. The same interfaces can also be listed by Interface Type. Web services that are currently listed in the Integration Repository can be invoked and consumed in BPEL Processes. The Open Interface tables or PL/SQL APIs can be invoked using the built-in Oracle Applications adapter.

Conclusion

Oracle BPEL Process Manager provides an open, standards-based platform to capture business processes in BPEL, protecting your intellectual property and investments while avoiding vendor lock-in. BPEL is to business process management what SQL was to data management. It is 60-80% less expensive to buy and maintain than traditional EAI (Enterprise Application Integration) solutions.

The BPEL Console reduces the cost and complexity of deploying and managing your business processes by enabling you to visually monitor the execution of each BPEL process, drill down into audit and view the details of each conversation or debug a running flow against its BPEL implementation.

Hope this article helps jumpstart introduction of Web services & BPEL in your organization, paving the way to adopt SOA across your enterprise!



Acknowledgement

Some of the content and images used in this article are courtesy of Oracle Corporation.

Related Web sites

- www.oracle.com/technology/bpel
- www.oracle.com/technology/tech/soa
- irep.oracle.com
- soasig.oaug.org

Basheer Khan, president and founder of Irvine, California-based Innowave Technology, is an Oracle ACE Director, Oracle Magazine's Integration Architect of the Year 2006 and Oracle Application Users Group (OAUG) Member of the Year in 2003. A frequent speaker on the capabilities of Oracle Applications & Technology at major conferences including OAUG COLLABORATE and Oracle OpenWorld, he has been published numerous times. He is the Executive Vice President of the OAUG Board of Directors and a member of the Fusion Council Steering Committee. Quite active in the Oracle community, he chairs the global Oracle Business Intelligence Special Interest Group (obisig.oaug.org). With 20 years of experience in information technology & solution architecture, his specialization includes enterprise/business process integration using SOA, Web Services and Oracle technologies. He received his bachelor's degree in mathematics from the University of Madras (India) and has completed post-graduate studies in computer science and applications at the College of Engineering, Anna University (India).