



P r o f e s s i o n a l   E x p e r t i s e   D i s t i l l e d

# Oracle SOA Infrastructure Implementation Certification Handbook (1Z0-451)

Successfully ace the 1Z0-451 Oracle SOA Foundation  
Practitioner exam with this hands on certification guide

**Kathiravan Udayakumar**

[www.it-ebooks.info](http://www.it-ebooks.info)

**[PACKT]** enterprise   
PUBLISHING professional expertise distilled

# Oracle SOA Infrastructure Implementation Certification Handbook (1Z0-451)

Successfully ace the 1Z0-451 Oracle SOA Foundation  
Practitioner exam with this hands on certification guide

**Kathiravan Udayakumar**



BIRMINGHAM - MUMBAI

# Oracle SOA Infrastructure Implementation Certification Handbook (1Z0-451)

Copyright © 2012 Packt Publishing

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor Packt Publishing, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

Packt Publishing has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, Packt Publishing cannot guarantee the accuracy of this information.

First published: May 2012

Production Reference: 1170512

Published by Packt Publishing Ltd.  
Livery Place  
35 Livery Street  
Birmingham B3 2PB, UK.

ISBN 978-1-84968-340-1

[www.packtpub.com](http://www.packtpub.com)

Cover Image by Mark Holland (MJH767@bham.ac.uk)

# Credits

**Author**

Kathiravan Udayakumar

**Project Coordinator**

Sai Gamare

**Reviewers**

Hans Forbrich

Arun Pareek

Jayaprakash Rajendran

Kalyan Sura

**Proofreaders**

Lisa Brady

Aaron Nash

**Indexer**

Tejal Daruwale

**Acquisition Editor**

Stephanie Moss

**Graphics**

Valentina D'silva

**Lead Technical Editor**

Dayan Hyames

Manu Joseph

**Production Coordinator**

Arvindkumar Gupta

**Technical Editors**

Manali Mehta

Ankita Shashi

Manmeet Singh Vasir

**Cover Work**

Arvindkumar Gupta

**Copy Editor**

Leonard D'Silva



# About the Author

**Kathiravan Udayakumar** has more than 8 years of IT experience with expertise in Oracle Fusion Middleware products and PeopleSoft applications. Interestingly, all his experiences relate to the world's first implementation (Greenfield) of IT system implementation and system integration.

Kathiravan holds a Bachelor's degree as a Gold Medalist in Electronics and Communication Engineering from Madurai Kamaraj University; a Master of Science degree in System Software Engineering from Birla Institute of Science and Technology, Pilani; and a Master of Business Administration degree in Sales and Marketing from Symbiosis Institute, Pune.

He has extensive experience in architecting and designing solutions using various Oracle Fusion and PeopleSoft products. He works for a highly-reputed IT consulting organization and is considered to be a key member of the Fusion CoE team. He has trained more than 100 associates in Fusion Middleware 11g to gain the skills for developing and designing solutions using Oracle SOA Suite 11g.

Kathiravan has extensive experience in the IT certification area for many years now, right from his college days. He holds various IT certifications from Cisco, IBM, Oracle, and BrainBench.

---

I would like to thank my family and friends who have always been supportive and helped me to move forward in all walks of my life. I would especially like to thank my colleague Jayaprakash Rajendran, who volunteered to review this book without any hesitation, and provided valuable comments on time despite his busy schedule. I would also like to thank all the members of Packt editorial team who have constantly helped me to improve this book with their valuable comments and service.

---

# About the Reviewers

**Hans Forbrich**, who is an Oracle ACE Director, has been working with Oracle technologies for a number of years, starting with Oracle Database v4 and Oracle Middleware using Oracle Web Server 3.0.

He is an Oracle University partner instructor; owner and principal partner in Forbrich Consulting Group, which is an OPN member. He is a consultant on high availability platforms using Oracle Linux and Oracle VM under Oracle Database and Oracle Fusion Middleware technologies, and is frequently invited to Oracle conferences to discuss a variety of Oracle-related topics.

**Arun Pareek** is an SOA Practitioner working on SOA-based implementation projects in the capacity of a Consultant and Architect for over 5 years now. He is also an IASA certified Software Architect and currently co-authoring a book on Oracle SOA Suite Administration for Packt Publishing.

He has been actively working on an SOA Suite of products for both BEA and Oracle including technologies like Service Bus, AIA, BPEL, BAM, BPA, and BPMN. He has a knack for designing systems that are scalable, performant, and fault-tolerant and is an enthusiast of Automated Continuous Integration techniques. He is also an active blogger on these technologies and runs a popular blog at <http://beatechnologies.wordpress.com>.

---

I would like to appreciate the encouragement that I received from my parents for helping me achieve many things in life. A special note of thanks to my wonderful wife, Karuna, for her constant support, cooperation, and patience, without which it would have been impossible for me to manage my work and life together.

---

**Jayaprakash Rajendran** has around 7 years of IT experience with expertise in PL/SQL programming, Oracle SOA 10g/11g Suite, PeopleSoft Application development, and Enterprise Application Integration with experience in Oracle SOA 11g, Oracle SOA Suite 10.1.3.x, and Oracle AIA.

He has experience in working with technology adapters such as File Adapter, FTP Adapter, Database Adapter, JMS Adapter, and Oracle AQ Adapter.

---

I would like to thank Kathiravan for giving me the opportunity to review this book and provide my input.

---

**Kalyan Sura** holds a Master's degree in Computers and has 15 years of IT experience. He provided technical and functional leadership on multiple large-scale full lifecycle ERP, CRM, and SOA implementations including formulating system scope and objectives.

Kalyan helped re-architect several mid-to-large scale B2B systems by providing detailed architectural specifications and a technical approach. He published and presented whitepapers on Oracle E-Business Suite and integration technologies at both regional and international Oracle conferences.

He recently completed a leadership course at Harvard BSP. He is well-known for his strategic thinking, technical expertise, implementation abilities, and innovation while handling tough assignments and providing quality results.

He successfully demonstrated applying existing and emerging technologies to deliver multiple concurrent large-scale projects. He is a persuasive negotiator successfully presenting complex tactical/program plans to corporate leaders and decision makers to achieve their buy-in.

---

I would like to thank Kathiravan and Packt Publishing for giving me an opportunity to be one of the reviewers of this book.

---

# www.PacktPub.com

## Support files, eBooks, discount offers and more

You might want to visit [www.PacktPub.com](http://www.PacktPub.com) for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at [www.PacktPub.com](http://www.PacktPub.com) and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at [service@packtpub.com](mailto:service@packtpub.com) for more details.

At [www.PacktPub.com](http://www.PacktPub.com), you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.



<http://PacktLib.PacktPub.com>

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

## Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

## Free Access for Packt account holders

If you have an account with Packt at [www.PacktPub.com](http://www.PacktPub.com), you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

## Instant Updates on New Packt Books

Get notified! Find out when new books are published by following @PacktEnterprise on Twitter, or the *Packt Enterprise* Facebook page.



# Table of Contents

<b>Preface</b>	<b>1</b>
<b>Chapter 1: Overview of Oracle SOA Certification</b>	<b>9</b>
Introduction to Oracle and the products	10
Oracle acquisition	11
Introducing Oracle SOA Suite 11g	13
New features in Oracle SOA Suite 11g	15
Oracle SOA certification objectives	15
Why get certified?	18
Exam details	18
Registration procedure	19
Summary	19
<b>Chapter 2: Service-Oriented Architecture Concepts</b>	<b>21</b>
What is SOA?	21
Do we really benefit a lot by designing applications using this method?	22
What is the business significance of this new architecture?	22
Do we need to learn new languages in doing this?	23
Describing the SOA concepts	23
Identifying standards that enable SOA	24
WS-Messaging standards	26
WS-Service Description and Discovery standards	26
WS-Reliability standards	27
WS-Transaction standards	27
WS-Security standards	27
WS-Business Process standards	28
WS-Management standards	28
<b>Describing the Event Driven Architecture (EDA)</b>	<b>29</b>
A real world example to understand EDA	29
EDA with Oracle SOA Suite 11g	30

---

Process followed in publishing and subscribing events	30
Features of Oracle SOA Suite 11g—EDN	30
<b>Reviewing the Service Component Architecture (SCA)</b>	<b>31</b>
SCA with Oracle SOA Suite 11g	34
<b>Summary</b>	<b>35</b>
<b>Self-review questions</b>	<b>35</b>
<b>Chapter 3: SOA Composite Application</b>	<b>39</b>
<hr/>	
<b>Oracle SOA Suite 11g components</b>	<b>40</b>
Oracle SOA component description: BPEL	40
Oracle SOA component description: Mediator	41
Oracle SOA component description: Business Rules	41
Oracle SOA component description: Human Task	42
Oracle SOA component description: Adapter	42
Oracle SOA component description: Business Activity Monitor	43
Oracle SOA component description: B2B	44
Oracle SOA component description: MDS	44
<b>SOA Composite Editor</b>	<b>44</b>
<b>Defining an Oracle SOA Composite application</b>	<b>48</b>
<b>Summary</b>	<b>56</b>
<b>Self-review questions</b>	<b>57</b>
<b>Chapter 4: Working with Adapters</b>	<b>59</b>
<hr/>	
<b>Introduction to Java Connector architecture</b>	<b>60</b>
<b>Adapter concepts and framework</b>	<b>61</b>
<b>Technology adapters: File, Database, JMS, and so on</b>	<b>63</b>
<b>Applications adapters: E-Biz Suite, PeopleSoft, Siebel, and SAP</b>	<b>68</b>
<b>Adapter design-time configuration</b>	<b>69</b>
File adapter configuration	70
FTP adapter configuration	76
MQ series adapter configuration details	80
Socket adapter service configuration	84
Direct Binding	87
HTTP binding wizard	88
<b>Explain the runtime configuration</b>	<b>90</b>
<b>Summary</b>	<b>91</b>
<b>Self-review questions</b>	<b>91</b>
<b>Chapter 5: Orchestrating Services with BPEL</b>	<b>95</b>
<hr/>	
<b>What BPEL helps to accomplish</b>	<b>96</b>
<b>Is BPEL a game changing technology?</b>	<b>97</b>
<b>Synchronous and asynchronous BPEL processes</b>	<b>97</b>
<b>Synchronous</b>	<b>98</b>



---

<b>Asynchronous – one-way</b>	<b>101</b>
<b>Asynchronous – bi-directional</b>	<b>102</b>
<b>BPEL components activities and Partner Links</b>	<b>105</b>
<b>Parallel flow and conditional branching</b>	<b>108</b>
<b>BPEL integration with Java</b>	<b>110</b>
<b>Summary</b>	<b>113</b>
<b>Self-review questions</b>	<b>114</b>
<b>Additional reference</b>	<b>116</b>
<b>Chapter 6: Advanced BPEL Concepts</b>	<b>117</b>
<b>Exception handling in composite applications</b>	<b>118</b>
System faults and custom faults	119
<b>The fault management framework</b>	<b>124</b>
The fault policy resolution process	127
<b>Compensation handling within a BPEL process</b>	<b>128</b>
<b>Correlation concepts</b>	<b>129</b>
Property and property alias	129
<b>Summary</b>	<b>132</b>
<b>Self-review questions</b>	<b>132</b>
<b>Chapter 7: Working with Mediator Concepts and Features</b>	<b>135</b>
<b>What the Mediator is not?</b>	<b>135</b>
<b>The Mediator component and its features</b>	<b>136</b>
<b>Creating and configuring a Mediator service component and creating Mediator routing rules</b>	<b>138</b>
Configuring a Mediator	140
Configuring a Mediator to subscribe to an Event	142
<b>Fault Management in a Mediator</b>	<b>143</b>
Mediator error groups	144
<b>Summary</b>	<b>145</b>
<b>Self-review questions</b>	<b>145</b>
<b>Chapter 8: Human Workflow</b>	<b>149</b>
<b>Oracle Human Workflow architecture</b>	<b>150</b>
Human Workflow service engine	151
Human Workflow services	151
Oracle Worklist Application	152
<b>Oracle Human Workflow features</b>	<b>154</b>
<b>Oracle Human Workflow concepts</b>	<b>155</b>
Participant	155
Task assignment	155
Routing policy	156
Ad-hoc routing	156

Task stakeholders	156
Task deadlines	157
Notifications	157
Task forms	157
Stages	158
Access rules	158
Standalone human tasks	158
Callbacks	158
Events	158
<b>Designing human tasks and services</b>	<b>158</b>
General	160
Data	160
Assignment	160
Deadline	163
Notification	164
Access level	165
Events	167
<b>Invoking a human task from a BPEL process</b>	<b>167</b>
Calling the human task from BPEL	172
<b>Summary</b>	<b>172</b>
<b>Self-review questions</b>	<b>172</b>
<b>Chapter 9: Business Rules</b>	<b>177</b>
<b>The Oracle Business Rules Architecture and Oracle Rule Engine details</b>	<b>179</b>
<b>The Rule Engine working model</b>	<b>181</b>
<b>Oracle Business Rules concepts</b>	<b>182</b>
<b>Creating rules in JDeveloper Rule Designer</b>	<b>188</b>
<b>Integrating simple Business Rules with the BPEL process</b>	<b>196</b>
Wiring the Business Rule component through SOA Composite	197
Including the Business Rule as an activity to the BPEL process	200
<b>Testing Business Rules from JDeveloper</b>	<b>201</b>
<b>Editing Business Rules from SOA Composer</b>	<b>204</b>
<b>Summary</b>	<b>205</b>
<b>Self-review questions</b>	<b>205</b>
<b>Chapter 10: Securing Services and Composite Applications</b>	<b>209</b>
<b>Web Services and composite security</b>	<b>210</b>
Transport-level security	211
Application-level security	211
XML Encryption (data confidentiality)	212
XML Signature	213
WS-Security	213

WS-Security tokens	213
WS-Policy	216
WS-SecurityPolicy	216
Web Services Addressing (WS-Addressing)	218
WS-ReliableMessaging	218
<b>The Fusion Middleware Security architecture</b>	<b>218</b>
<b>Security policy management and identity propagation</b>	<b>219</b>
<b>OWSM Agents</b>	<b>224</b>
<b>Summary</b>	<b>225</b>
<b>Self-review questions</b>	<b>226</b>
<b>Chapter 11: SOA Integration Points</b>	<b>229</b>
<b>The Oracle Service Bus concepts and architecture</b>	<b>230</b>
OSB product architecture	230
<b>Oracle Business Activity Monitoring</b>	<b>232</b>
The Oracle BAM architecture	232
The Oracle BAM server	233
The Oracle BAM web applications	233
The Oracle BAM ADF control	233
Integrating BAM with Oracle SOA Suite	233
<b>Oracle B2B</b>	<b>234</b>
Features of an Oracle B2B product	234
Industry-standard protocols supported by Oracle B2B	235
Oracle B2B 11g home page	236
Oracle B2B deploy agreement	237
Oracle B2B metrics report	237
Oracle B2B Reference Integration Architecture	238
<b>Summary</b>	<b>238</b>
<b>Self-review questions</b>	<b>239</b>
<b>Chapter 12: Monitoring and Managing SOA 11g Deployment</b>	<b>245</b>
<b>Deploying and undeploying the SOA composite application</b>	<b>247</b>
Using JDeveloper to deploy the composites	247
Using the SOA EM SOA deployment wizard to deploy the SAR	
file generated through JDeveloper	251
Generating the SAR file through JDeveloper	251
Deploying the SAR file from the console	254
Undeploying composites	259
SOA configuration plan	260
<b>Management of the SOA composite application using the</b>	
<b>Enterprise Manager</b>	<b>264</b>
SOA farms and topology features	268
Managing and monitoring the SOA composite	270
SOA EM Export features	275

*Table of Contents*

---

Export options	275
Monitoring SOA Service Engines	276
Engine-level Dashboard view	277
BPEL Engine Statistics	278
Monitoring SOA performance	279
Managing soa-infra and Service Engine properties	281
<b>The Enterprise Manager Grid Control in the SOA Management Packs</b>	<b>287</b>
<b>Summary</b>	<b>288</b>
<b>Self-review questions</b>	<b>289</b>
<b>Additional references</b>	<b>292</b>
<b>Appendix A: Pre-assessment Test</b>	<b>293</b>
<b>Appendix B: Post-assessment Test</b>	<b>299</b>
<b>Appendix C: Exam Reference</b>	<b>311</b>
SOA concepts	311
SOA composite applications	312
Working with adapters	313
Orchestrating services with BPEL	314
Advanced BPEL concepts	317
Mediator	318
Business Rules	320
Human workflow	321
Monitoring and managing SOA 11g deployment	321
SOA 11g integration points	321
Miscellaneous	322
<b>Appendix D: Answers</b>	<b>325</b>
Chapter 2	325
Chapter 3	326
Chapter 4	327
Chapter 5	327
Chapter 6	328
Chapter 7	329
Chapter 8	330
Chapter 9	331
Chapter 10	332
Chapter 11	332
Chapter 12	334
Appendix A	335
Appendix B	336
<b>Index</b>	<b>341</b>

---

# Preface

Certification is the first big step towards getting yourself recognized in the extremely large developer community. It demonstrates that you have a working knowledge, and gives you an edge in the market space and employment space. It helps you to get a better salary through promotions or increments in a few cases.

Completing the Oracle SOA Infrastructure Implementation Certification develops your conceptual and real-world understanding of the primary components of Oracle SOA Suite, including BPEL process engine and Oracle Service Bus, and will allow you to become familiar with the Service Oriented Architecture concepts.

This book guides you through the prescribed syllabus for the 1Z0-451 Oracle SOA Foundation Practitioner exam. This definitive certification guide provides a disciplined approach to be adopted for successfully clearing the 1Z0-451 Oracle SOA Foundation Practitioner exam to attain the Oracle Service Oriented Architecture Infrastructure Implementation Certified Expert title.

This book will introduce you to essential SOA concepts that will enable and help you to identify the standards that enable SOA; understand the Service Component Architecture; describe the Event Driven Architecture; and help you to identify the difference between EDA and SOA. In the mid-set of the book, you will understand some of the core concepts in Oracle SOA Suite that help you to work with Adapters, BPEL, Mediator, Human Workflow, and Business Rules Components. Towards the end of the book, readers will be able to understand the techniques to secure the services and integrate the Oracle SOA Suite Components with other SOA integration points such as BAM, B2B, and OSB. Pre-assessment and Post-assessment sections will help to test the preparedness of the readers for the exam. Don't miss the Exam Reference section before you begin the exam.

## What this book covers

*Chapter 1, Overview of Oracle SOA Certification*, will help you to understand exam objectives, exam preparation methods, and the exam registration procedure.

*Chapter 2, Service-Oriented Architecture Concepts*, will help you to understand the following list of exam objectives:

- Describing the SOA concepts
- Identifying standards that enable SOA
- Describing the Event Driven Architecture (EDA)
- Reviewing the Service Component Architecture (SCA)

*Chapter 3, SOA Composite Application*, will help you to understand the following list of exam objectives:

- Describing Oracle SOA Suite 11g components
- Describing the Service components
- Defining a composite application
- Describing SOA Composite Editor

*Chapter 4, Working with Adapters*, will help you to understand the following list of exam objectives:

- Describing adapter concepts and framework
- Describing technology adapters: File, Database, JMS, and so on
- Describing applications adapters: EBiz suite, PeopleSoft, Siebel, and SAP
- Explaining adapter runtime configuration
- Explaining adapter design-time configuration

*Chapter 5, Orchestrating Services with BPEL*, will help you to understand the following list of exam objectives:

- Explaining BPEL components, activities, and Partner Links
- Describing synchronous and asynchronous BPEL processes
- Describing message transformations and XSLT
- Explaining parallel flow and conditional branching
- Explaining BPEL integration with Java

*Chapter 6, Advanced BPEL Concepts*, will help you to understand the following list of exam objectives:

- Describing exception handling in composite applications
- Describing the fault management framework
- Describing compensation handling within a BPEL process
- Describing correlation concepts

*Chapter 7, Working with Mediator Concepts and Features*, will help you to understand the following list of exam objectives:

- Explaining the Mediator component and its features
- Creating and configuring a Mediator service component
- Creating Mediator routing rules

*Chapter 8, Human Workflow*, will help you to understand the following list of exam objectives:

- Describing Human Workflow concepts, features, and architecture
- Designing human tasks and services
- Invoking a human task from a BPEL process

*Chapter 9, Business Rules*, will help you to understand the following list of exam objectives:

- Explaining business rule concepts
- Describing the Oracle Business Rules architecture
- Describing Oracle Rules Engine
- Creating rules with the JDeveloper Rules Designer
- Integrating a simple rule with a BPEL process

*Chapter 10, Securing Services and Composite Applications*, will help you to understand the following list of exam objectives:

- Explaining Web Services and composite security
- Describing the Fusion Middleware security architecture
- Understanding security policy management and identity propagation
- Describe OWSM agents



*Chapter 11, SOA Integration Points*, will help you to understand the following list of exam objectives:

- Describing Oracle Service Bus concepts and architecture
- Describing Oracle Business Activity Monitoring
- Describing Oracle B2B

*Chapter 12, Monitoring and Managing SOA 11g Deployment*, will help you to understand the following list of exam objectives:

- Describing deploy and un-deploy SOA composite applications
- Describing management of SOA composite applications using the Enterprise Manager
- Describing Enterprise Manager Grid Control SOA Management Packs

*Appendix A, Pre-assessment Test*, will provide you with an opportunity to test your knowledge in Oracle SOA Suite before going through this book.

*Appendix B, Post-assessment Test*, will provide you with an opportunity to test your knowledge in Oracle SOA Suite. We have provided full-length test papers in this chapter for an exam experience.

*Appendix C, Exam Reference*, will provide key points to be remembered for the exam, and other additional details that are required to understand Oracle SOA Suite 11g in detail.

*Appendix D, Answers*, contains answers and explains the questions found at the end of each chapter and appendix.

## **What you need for this book**

Very basic understanding of Service Oriented Architecture and prior introductory knowledge of Oracle SOA Suite 11g is desirable.

## **Who this book is for**

This book is for SOA architects, technical consultants, application developers, and analysts who want to successfully clear the 1Z0-451 Oracle SOA Foundation Practitioner exam to attain the Oracle SOA Infrastructure Implementation Certification.

## Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text are shown as follows: "The most commonly occurring system faults are `remoteFault` and `bindingFault`."


A block of code is set as follows:


```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Generated by Oracle SOA Modeler version 1.0 at [12/28/10 7:03
PM]. -->
<composite name="ProcessClaimsInfoApps"    revision="1.0"
label="2010-12-28_19-03-39_049"
        mode="active"        state="on"    xmlns="http://xmlns.oracle.
com/sca/1.0"
        xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
        xmlns:orawsp="http://schemas.oracle.com/ws/2006/01/policy"
        xmlns:ui="http://xmlns.oracle.com/soa/designer/">
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items are set in **bold**:

```
<wsdl:portType name="SyncBPELProcess">
  <wsdl:operation name="process">
    <wsdl:input message="client:SyncBPELProcessRequestMessage"/>
    <wsdl:output message="client:SyncBPELProcessResponseMessage"/>
    <wsdl:fault message="client:FaultMessage"
      name="FaultMessage"/>
  </wsdl:operation>
</wsdl:portType>
```

**New terms** and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "Keep the **Editable via worklist** checkbox as checked and click **OK**".

 Warnings or important notes appear in a box like this.

 Tips and tricks appear like this.

## Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book – what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to [feedback@packtpub.com](mailto:feedback@packtpub.com), and mention the book title through the subject of your message.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on [www.packtpub.com/authors](http://www.packtpub.com/authors).

## Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

## Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books – maybe a mistake in the text or the code – we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting <http://www.packtpub.com/support>, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded to our website, or added to any list of existing errata, under the Errata section of that title.

## Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at [copyright@packtpub.com](mailto:copyright@packtpub.com) with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

## Questions

You can contact us at [questions@packtpub.com](mailto:questions@packtpub.com) if you are having a problem with any aspect of the book, and we will do our best to address it.



# 1

## Overview of Oracle SOA Certification

Welcome to the new world of Oracle. You have picked the right book for gaining expertise in Oracle SOA Suite and certifying yourself as a highly qualified professional in the growing pool of the Oracle SOA Consultant/Developer community. This book will help you to understand the exam objectives for getting the Oracle Service Oriented Architecture Infrastructure Implementation Certified Expert status through the 1Z0-451 Oracle SOA Foundation Practitioner exam. This book will provide you with real-world scenarios for explaining the concepts involved in using Oracle SOA components. The focus of this exam is on the Oracle SOA Suite 11g release.

**Service Oriented Architecture** is growing in importance day-by-day in the information technology space, and the number of customers implementing SOA solutions is growing as well. Oracle SOA Suite has emerged as a strong player in this field and it is the number 1 ranked middleware product today in the market that supports Open standards, hot pluggable, easy to develop and deploy integration components in agile and nimble fashion. Having expertise in this product will give a bigger edge to your career and provide ample opportunities in an ever changing Oracle World.

Before getting deep into the concepts, this chapter will provide you with an overview of Oracle – its products, its technologies, certification, its benefits, and importance.

In this chapter, we shall attempt to understand the following:

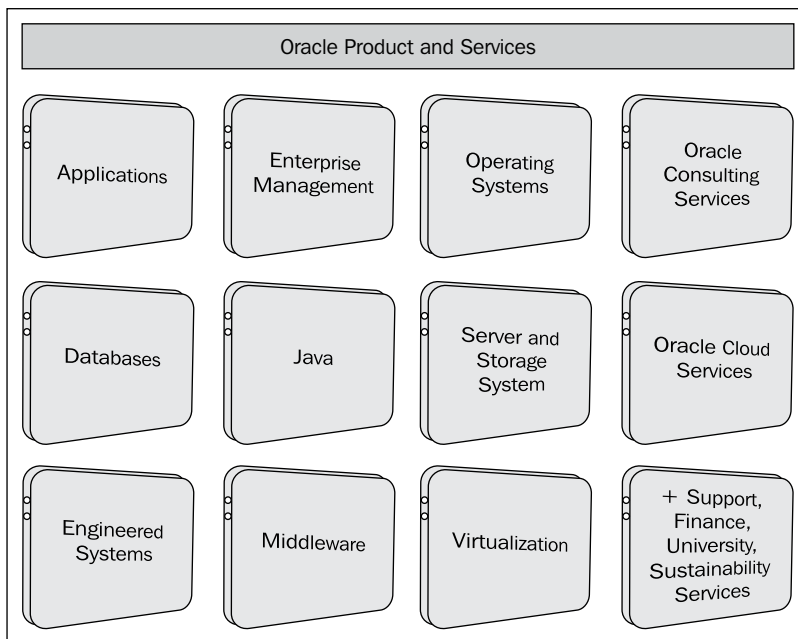
- Oracle and products
- Oracle SOA Suite 11g
- Oracle SOA certification objectives
- Why get certified?

- Exam details
- Exam registration procedure

## Introduction to Oracle and the products

Most of us should be knowing Oracle as a leading database vendor for many years. However, it has also emerged as a company focused on hardware, software, middleware, solutions, and services by acquiring a large number of companies. Their product catalog now ranges from packaged implementations to solutions and services, middleware, and database products among many industry solutions.

The following figure illustrates it in detail:





## Oracle acquisition

Right after acquisition of PeopleSoft in 2005, Oracle never looked back from the acquisition stream, which can be depicted from the following details:

Product Group	Product Name/Company (Month and Year of Acquisition)
Database	* DataScaler (October 2010)
	* e-Test (acquired from Empirix) (March 2008)
	* Innobase (October 2005)
	* Moniforce (December 2007)
	* mValent (February 2009)
	* Secerno (May 2010)
	* Sleepycat (February 2006)
	* TimesTen (June 2005)
	* TripleHop (June 2005)
Middleware	* AmberPoint (February 2010)
	* BEA (January 2008)
	* Bharosa (July 2007)
	* Bridgestream (September 2007)
	* Captovation (January 2008)
	* ClearApp (September 2008)
	* Context Media (July 2005)
	* HyperRoll (September 2009)
	* GoldenGate (July 2009)
	* Java (April 2009)
	* Oblix (March 2005)
	* OctetString (November 2005)
	* Passlogix (October 2010)
	* Sigma Dynamics (August 2006)
	* Stellent (November 2006)
	* Sunopsis (October 2006)
	* Tacit Software (November 2008)
	* Tangosol (March 2007)
	* Thor Technologies (November 2005)

Product Group	Product Name/Company (Month and Year of Acquisition)
<b>Applications</b>	<ul style="list-style-type: none"><li>* AppForge (April 2007)</li><li>* Haley (October 2008)</li><li>* Interlace Systems (October 2007)</li><li>* LogicalApps (October 2007)</li><li>* Market2Lead (May 2010)</li><li>* Silver Creek Systems (January 2010)</li><li>* TempoSoft</li><li>* Agile (May 2007)</li><li>* ATG (November 2010) (pending)</li><li>* Hyperion (March 2007)</li><li>* PeopleSoft (January 2005)</li><li>* Primavera (October 2008)</li><li>* Siebel (January 2006)</li><li>* Telephony@Work (June 2006)</li></ul>
<b>Server and Storage</b>	<ul style="list-style-type: none"><li>* Sun (April 2009)</li><li>* Virtual Iron (May 2009)</li></ul>
<b>Industry Solutions</b>	<p>Communications and Media</p> <ul style="list-style-type: none"><li>* Convergin (February 2010)</li><li>* eServGlobal's Universal Service Platform (USP) (May 2010)</li><li>* HotSip (February 2006)</li><li>* MetaSolv Software (October 2006)</li><li>* Net4Call (April 2006)</li><li>* Netsure Telecom Limited (September 2007)</li><li>* Portal Software (April 2006)</li><li>* Sophoi (October 2009)</li></ul> <p>Engineering and Construction</p> <ul style="list-style-type: none"><li>* Primavera (October 2008)</li></ul> <p>Financial Services</p> <ul style="list-style-type: none"><li>* i-flex (August 2005)</li></ul> <p>Health Sciences</p> <ul style="list-style-type: none"><li>* Phase Forward (April 2010)</li><li>* Relsys (March 2009)</li></ul> <p>Industrial Manufacturing</p> <ul style="list-style-type: none"><li>* Agile (May 2007)</li><li>* Conformia Software (June 2009)</li><li>* Demantra (June 2006)</li><li>* G-Log (September 2005)</li></ul>

---

Product Group	Product Name/Company (Month and Year of Acquisition)
	Insurance
	* AdminServer (May 2008)
	* Skywire Software (June 2008)
	Retail
	* 360Commerce (January 2006)
	* Advanced Visual Technology (AVT) (October 2008)
	* ProfitLogic (July 2005)
	* Retek (April 2005)
	Utilities
	* LODESTAR (April 2007)
	* SPL WorldGroup (November 2006)

With all of this acquisition, Oracle SOA Suite is gaining the importance for integrating the applications for completing business process-oriented fusion applications. Oracle SOA Suite will continue to evolve as the products are acquired by Oracle and as customers tend to use them in a standalone middleware context.

## Introducing Oracle SOA Suite 11g

Oracle SOA Suite 11g is a very critical release of the Oracle middleware product set where there is a significant change from the previous release. Oracle SOA Suite 11g has significant changes to its architecture used in the product; it follows SCA (Service Component Architecture) for building Oracle SOA Components, where the components can be integrated using specifications provided by the Service Component Assembly. In Oracle SOA Suite 10g, the components are developed individually and they are integrated using the service calls or native API calls if the service engines are accessible in the same runtime. But Oracle SOA Suite 11g follows a unique style of developing, deploying, and runtime components. The following table shows the major differences between Oracle SOA Suite 10g and 11g:

Topic	10g	11g
Application Server	Oracle Application Server	Web logic
Service Bus	ESB	Mediator (Note: This will only mediate the request between the Oracle SOA Components in Composite)

<b>Topic</b>	<b>10g</b>	<b>11g</b>
Orchestration Engine	BPEL	BPEL – new activities such as create, bind, remove entity and signals
SDO Support	Not available	Available in BPEL
Component Architecture	Not available	SCA
Large File Handling	Not available	Available
Scalable DOM Support	Not available	Available
Logging Support	Custom code	Through policy
DVM	Available in ESB Console	Can be created from JDeveloper
Cross Reference	Available through command-line tool	Can be created from JDeveloper
Security	OWSM is used	Security policies can be used
Business Rules	Business Rule Console is used to update and edit it	Rules can be created and updated using JDeveloper
Events	Only Pick Pattern Supported Events in BPEL	EDN Framework available for Publishing and Subscribing to events
End-End Tracking of Application	Not available	ECID helps to track the instances end-to-end
Fault-Policies	Only BPEL Faults are supported	Mediator and BPEL Faults can be handled
Streaming Support in Adapter	Not available	Available
Runtime	Diversified	Unified
Transformation – Multiple Sources	Not available	Available
BAM	Sensor Framework is used to integrate with BPEL	Sensor Framework + BAM adapter are provided

## New features in Oracle SOA Suite 11g

The new features included in Oracle SOA Suite 11g are as follows:

Topics	New Features
<b>Mediator</b>	Sequential and Parallel Routing Schematron-based Validation Support for All MEP Patterns Assigning values to JCA Properties Ability to publish events using EDN Ability to subscribe to events using EDN
<b>BPEL</b>	Ability to bind, create, and remove entity objects in ADF-BC partner links
<b>EDN</b>	Business Events are defined using the event definition language
<b>Enterprise Manager</b>	End-to-end instance tracking can be done using ECID

## Oracle SOA certification objectives

This section will provide you with a list of objectives that are to be covered in the Oracle Service Oriented Architecture Infrastructure Implementation Certified Expert Examination. This book has been arranged in a fashion such that the topics are in-line with the exam objectives.

Topic ID	Topic Name	Objective ID	Objective Description
1	<b>Service-Oriented Architecture Concepts</b>	1.1	Describes Service-Oriented Architecture (SOA) concepts
		1.2	Identifies standards that enable SOA
		1.3	Reviews Service Component Architecture (SCA)
		1.4	Describes Event-Driven Architecture (EDA)
2	<b>SOA Composite Applications</b>	2.1	Describes Oracle SOA Suite 11g components
		2.2	Describes the service components
		2.3	Defines a composite application
		2.4	Describes the SOA composite editor

<b>Topic ID</b>	<b>Topic Name</b>	<b>Objective ID</b>	<b>Objective Description</b>
3	<b>Adapters</b>	3.1	Describes adapter concepts and framework
		3.2	Describes technology adapters: File, Database, JMS, and so on
		3.3	Describes Applications Adapters EBiz suite, PeopleSoft, Siebel, and so on
		3.4	Explains adapter runtime configuration
		3.5	Explains adapter design-time configuration
4	<b>Orchestrating Services with BPEL</b>	4.1	Explains BPEL components activities and partner links
		4.2	Describes synchronous and asynchronous BPEL processes
		4.3	Describes message transformations and XSLT
		4.4	Explains parallel flow and conditional branching
		4.5	Explains BPEL integration with Java
5	<b>Advanced BPEL Concepts</b>	5.1	Describes exception handling in composite applications
		5.2	Describes the Fault Management Framework
		5.3	Describes compensation handling within a BPEL process
		5.4	Describes correlation concepts
6	<b>Working with Mediator Components</b>	6.1	Explains the Mediator component and its features
		6.2	Creates and configures a Mediator service component
		6.3	Creates Mediator routing rules
7	<b>Human Workflow</b>	7.1	Describes Human Workflow concepts, features, and architecture
		7.2	Designs Human Tasks and Services
		7.3	Invokes a Human Task from a BPEL process

Topic ID	Topic Name	Objective ID	Objective Description
8	<b>Oracle Business Rules Concepts</b>	8.1	Explains business rules and concepts
		8.2	Describes the Oracle Business Rules architecture
		8.3	Describes Oracle Rules Engine
		8.4	Creates a rule with the JDeveloper Rules Designer
		8.5	Integrates a simple rule with a BPEL process
9	<b>Secure services and Composite Applications</b>	9.1	Explains Web Services and Composite Security
		9.2	Describes the Fusion Middleware security architecture
		9.3	Understands Security Policy Management, Identity Propagation
		9.4	Describes OWSM Agents
10	<b>SOA 11g Integration Points</b>	10.1	Describes Oracle Service Bus Concepts and Architecture
		10.2	Describes Oracle Business Activity Monitoring
		10.3	Describes Oracle B2B
11	<b>Monitoring and managing SOA 11g Deployment</b>	11.1	Describes deploy and undeploy of an SOA Composite application
		11.2	Describes management of an SOA Composite application using the Enterprise Manager
		11.3	Describes Enterprise Manager Grid Control SOA Management Packs



## Why get certified?

Getting certified in the IT profession provides a greater advantage in many ways. Most of the organizations in recent times have started to insist on the importance of learning as one of the key goals in professional growth. In certain organizations today, award points are provided and considered for promotions from the current level.

Certification will provide an opportunity to gain complete knowledge about the product beyond the simple real world practical implementation techniques that are being followed.

Oracle Justification for getting certified can be obtained from the following link: [http://education.oracle.com/pls/web\\_prod-plq-dad/db\\_pages.getpage?page\\_id=73](http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=73)

## Exam details

Exam Number: 1Z0-451

Exam Name: Oracle SOA Foundation Practitioner Duration: 90 minutes

Number of Questions: 60

Exam Price: US\$ 195

Pricing may vary slightly by country or by localized currency. Check Pearson VUE's website for the exact pricing in your country.

Passing: 63% (Base %) Passing Score is subjected to change.



Please visit the following links to know more details about the exam  
<http://www.oracle.com/partners/en/knowledge-zone/middleware/oracle-service-oriented-architecture-soa/soa-exam-page-170307.html>  
<http://www.oracle.com/us/education/certification/score-info-079154.html>

## **Registration procedure**

The registration procedure for taking the Oracle Certification Exam can be obtained from the following link:

<http://www.pearsonvue.com/Oracle/>

## **Summary**

In this chapter, we have provided you with an overview of the Oracle SOA Certification Exam with the required information to understand the importance of the product and need for you to get certified with this next generation middleware technology.

We are confident that you are convinced to take the certification and make this enjoyable journey with us through next set of chapters.



# 2

## Service-Oriented Architecture Concepts

**Service-Oriented Architecture (SOA)** has been a buzz word in the IT industry for a few years now. What is SOA? Do we really benefit a lot by designing applications using this method? What is the business significance of this new architecture? Do we need to learn new languages in doing this? These are the questions we will discuss before getting deeper into the following certification objectives:

- Describing the SOA concepts
- Identifying standards that enable SOA
- Describing the **Event Driven Architecture (EDA)**
- Reviewing the **Service Component Architecture (SCA)**

### What is SOA?

SOA can have different definitions based on the context in which it is viewed and applied:

- **Application integration context:** SOA is a principle by which applications are integrated seamlessly using the services exposed from the application, using a standard plug-and-play model
- **Application design context:** SOA is a principle by which the functionalities of the applications are exposed as a service to be consumed by the applications to extend and reuse their features

- **Composite application context:** SOA is a principle by which distributed systems are built together to form a **composite** large system with fine-grained and coarse-grained services from different layers of the enterprise applications
- **Business process orchestration context:** SOA is a principle by which the business process automations are built using the services exposed from the application to complete a business process transaction

## Do we really benefit a lot by designing applications using this method?

Reusability and loose coupling are the keys to designing applications in an object-oriented world; this holds true in a service orientation one too, and they are the corner stones for building SOA. IT has progressed over the years from creating languages for specific domains to generic purpose languages. This had led to applications being developed using different languages in different platforms. With the advent of **Java virtual machine (JVM)**, the constraints to recode the applications to various operating systems were removed, which provided platform independency, but still the question of applications communicating with each other, written in different languages, existed before service orientation.

Designing the application in a service orientation way helps the application functionalities to be exposed to other applications and allow applications to be accessed from the external applications or the integration components in a standard way.

## What is the business significance of this new architecture?

After the introduction of the Internet, businesses are going online; dependency on the Internet for business is also increasing. The distributed operations are becoming more significant with geo-political situations. Competitiveness in the market to provide best-of-breed products and solutions drives the organization to pull the best of the best from different parts of the world. This requires IT systems to be implemented in different parts of the world using different technologies and solutions. As we are all aware, business departments cannot run in silos, neither are they created for that purpose; they need to work together to complete a business process function. Having said this, IT applications have to communicate or talk to each other and expose the services of each application to be consumed by another to complete the business process. SOAs way of developing and integrating applications will provide lots of benefits to business. We can identify the benefits by taking a look at the following example.

Marw, an infrastructure company, was running its operations using legacy applications, and it integrates applications using the traditional way by sharing the application using flat files. Sharing the flat files with other applications happens through nightly run jobs, which created a significant delay in making decisions for various business units as there is latency in making the data available to the decision makers. This leads to loss of business or a delay in responding to the customer's needs.

Marw decided to implement **Oracle SOA Suite** to integrate applications, so that information could be available to decision makers as the data changes in the applications. Marw saw a major change in efficiency of the operations in business. This led to a very significant advantage in running the business. SOA can help businesses to improve their business efficiency in integrating the applications in a real-time manner using components such as **BPEL (Business Process Execution Language)** and **ESB (Enterprise Service Bus)**.

## Do we need to learn new languages in doing this?

Yes, traditional programmers of Java or C# have to learn a few more technologies to get them to adapt to SOA. Conceptual understanding of XML, web services, **Business Process Management (BPM)**, ESB, and BPEL are key to understanding and implementing the solutions using SOA.

## Describing the SOA concepts

We have discussed in detail what SOA is, its benefits, its significance, and so on, in the previous sections of this chapter. In this section, we will discuss the concepts that enable SOA.

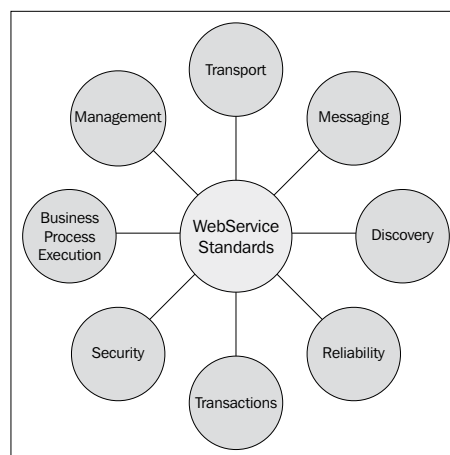
SOA is a broad term and concepts involved in enabling the technology require deeper understanding from various perspectives, which are described as follows:

- **Service design:**
  - **Interoperability:** Ability for the technology to interoperate with various application standards is one of the inherent characteristics required for enabling SOA. **Web Services Description Language (WSDL)** helps to achieve these characteristics for enabling interoperability among various enterprise applications.

- **Loose coupling:** The ability to independently operate the functions with minimal dependency is a characteristic that needs to be considered during the service design. Loose coupling can be achieved through asynchronous interaction patterns among applications. **Java Message Service (JMS)** and **Oracle Advanced Queuing (AQ)** can help in implementing such patterns in SOA.
  - **Reusability:** The ability to reuse the services among different application's functionality is required for the components to interact with the each other. This characteristic is also required for building composite applications, where the services from different enterprise applications are composed together.
  - **Granularity:** A service should be defined with a required level of granularity. Services should be designed at fine-grain or coarse-grain level, based on the applications of the service and the layer in which the service will operate. This is an important characteristic of a service that needs to be identified during the service requirement definition phase.
- **Service management:**
    - **Versioning of services:** Versioning of services is required for the services to be consumed and managed in the right way.

## Identifying standards that enable SOA

Standards that enable SOA are not evolved in a big bang model; they evolved over a period of time and they contribute towards the SOA standards and their improvement. The following diagram shows various standards that contribute towards SOA:



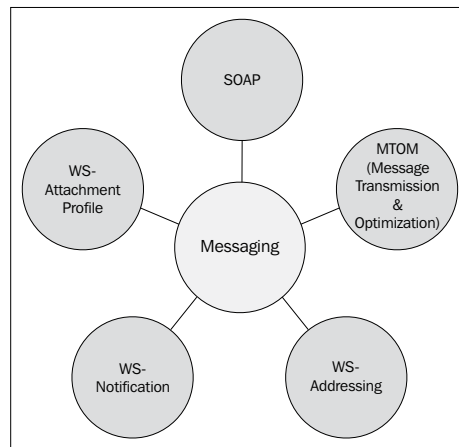
The following table describes the important standards that enable SOA:

Sr. no.	Standard	Standard Description
1	WSDL	<b>Web Services Description Language</b> is an XML-based language used to describe the service exposed by the application.
2	SOAP	<b>Simple Object Access Protocol</b> is a platform-neutral transport protocol used to access remote service.
3	MTOM	<b>Message Transmission Optimization Mechanism</b> is a method of efficiently sending binary data to and from web services. MTOM when used with XOP, is used to transmit larger messages.
4	XOP	<b>XML-binary Optimized Packaging</b> is a means of more efficiently serializing XML infosets that have certain types of content. This is used to optimize the XML data manipulation.
5	UDDI	<b>Universal Description, Discovery, and Integration</b> is a specification provided to store the service information available in the organization. This is a registry specification that facilitates the registry, discovery, and integration of services in a standard XML-based implementation.
6	WS-Coordination	<b>WS-Coordination</b> is the protocol used to describe the context that needs to be coordinated between different services.
7	WS-Security	<b>Web Services Security</b> is a specification for applying security on the web services.
8	WS-ReliableMessaging	<b>WS-ReliableMessaging</b> is a specification to describe the reliable delivery of messages to different services or between services.
9	BPEL4WS	<b>BPEL4WS</b> is a new standard defined to execute the business process using the web service orchestration model.
10	XML	<b>Extensible Markup Language</b> is a standard used to represent the business data in the industry standard manner to exchange information among the applications.
11	XSD	<b>XML Schema Definition</b> is a standard to define the structure of an XML document.
12	XSLT	<b>Extensible Stylesheet Language Transformations</b> is a standard used to describe the data transformation from one data format to another.

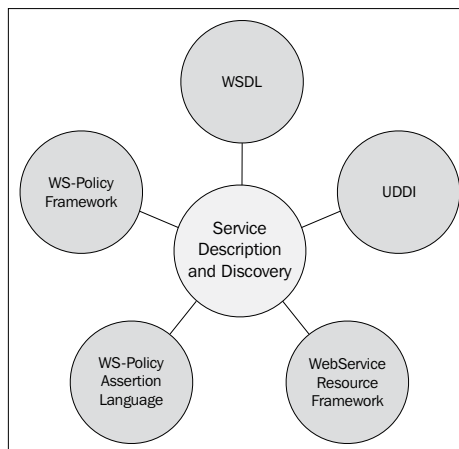


The following diagrams show the exhaustive list of web service standards that enable SOA.

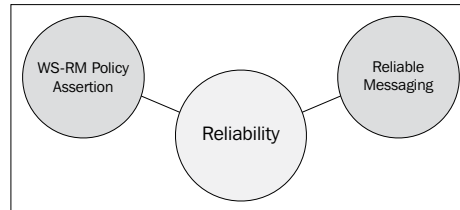
## **WS-Messaging standards**



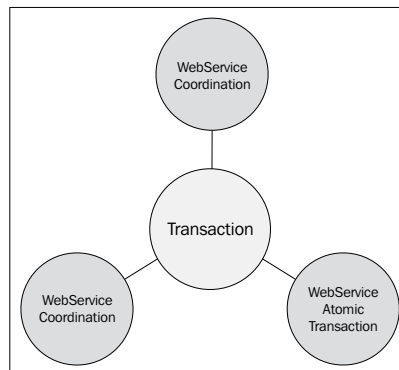
## **WS-Service Description and Discovery standards**



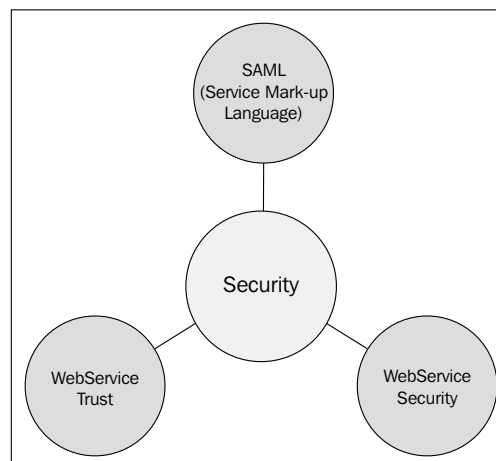
## WS-Reliability standards



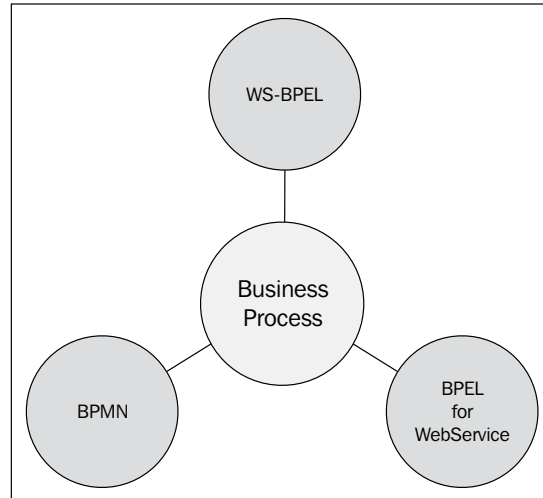
## WS-Transaction standards



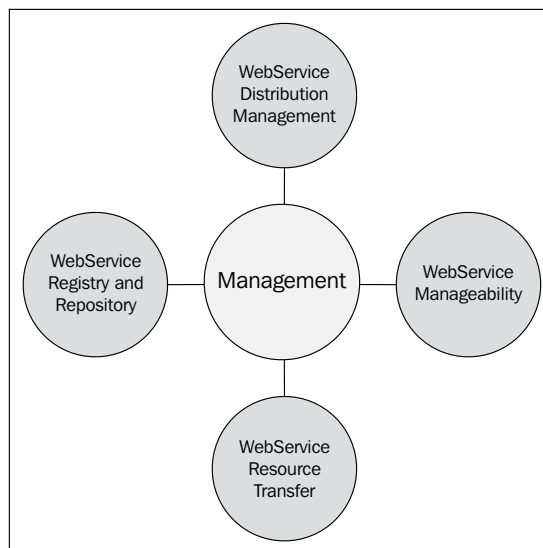
## WS-Security standards



## **WS-Business Process standards**



## **WS-Management standards**



## **Describing the Event Driven Architecture (EDA)**

EDA is a way to design enterprise application integration using the events generated from the applications. **Event** is the information generated from the business application for the business event that occurred. Events generated from the application will be the driver for business process completion or for notification of the information, regarding the flow of an event in the enterprise.

EDA is the way by which the applications are integrated in an asynchronous way. The provider and consumer of the event need not know the location of the service, as in SOA. Event manager will take care of publishing and subscribing the events generated by the application. The provider and consumer of the applications need to know only the event name and event data to be published. From the above discussion, it may hit some of us that this can be achieved using a queue-based architecture. The major difference between EDA and queue integration-based architecture is the degree of decoupling provided by EDA versus queue. As specified earlier, EDA doesn't force the provider and consumer of the data to know the location. An environment that supports this style of integration has a coordinator that takes care of the location transparency.

## **A real world example to understand EDA**

In a business world, change keeps happening every minute of every day, but the changes are not synchronized at an equal rate in all the business applications that host different services, which creates a disparity in the information being available to different business users, with the traditional integration model or even with real-time sync-based model. The initiator of the change has to take care of updating all the required systems, but the addition and removal of a system keeps happening throughout the clock in larger enterprises. For the systems to asynchronously send information to various systems, an integration layer is required for the propagation information. EDA provides the required specifications, tools, and technique to achieve this.

For example, if you would like to send some personal information updated in a self-service application to various applications, EDA would be the best choice. Events are defined and published using the tools available in the system.

## EDA with Oracle SOA Suite 11g

Oracle SOA Suite 11g has an additional engine/layer called **business event**, which takes care of supporting EDA. From a business event layer, publication and subscription of the events can happen. This is done through **Event Delivery Network (EDN)**. Oracle SOA Suite 11g supports publishing and subscribing to events for the following list of components:

Sr. no.	Component	Publishing events	Subscribing events
1	Mediator	Yes	Yes
2	BPEL	Yes	Yes
3	Business Rules	N/A	N/A
4	Oracle Service Bus	N/A	N/A
5	Human Workflow	N/A	N/A

## Process followed in publishing and subscribing events

The following process is followed in publishing and subscribing events:

1. Define the schema to be used in publishing the data for the event.
2. Define the event to be published/subscribed in an event definition file (\*.edl).
3. Deploy the event definition file to a common location.
4. Define the component that will publish the event by connecting through or accessing the system and deploy the component.
5. Create the subscription components and deploy the component.

## Features of Oracle SOA Suite 11g—EDN

The following are the list of features available as a part of EDN:

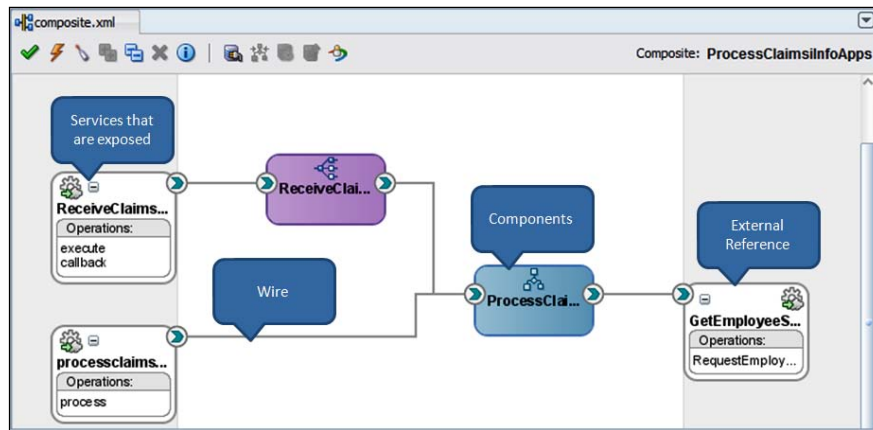
- Ability to publish and subscribe events across SOA suite environments
- Ability to monitor the faulted subscription or published events
- Ability to monitor the events published and subscribed
- Ability to test the publisher components
- Ability to view the event definition from the console

## Reviewing the Service Component Architecture (SCA)

Componentization is a gift to software developers. It helps the application components to reuse the functionalities across various modules. SCA is a new specification provided by OASIS in assembling the service components together.

SCA applications contains various components that can be wired together to build a composite application. Each and every component can be of a different technology. Currently, SCA specification supports Java (POJO), C++, PHP, BPEL, EJB, and Spring. SCA defines a layer called composite, where different components can be combined together and wired.

The following screenshot shows the anatomy of a composite:



A component will hold the business logic of the system being implemented. Services are exposed to the external world to access the components. There can be any number of services that can be exposed. Implementation of the services differs based on the technology being used to implement the component. If the component is implemented using Java, a service could be of a simple Java interface, whereas BPEL could expose the service using the WSDL. The term *service* used in SCA architecture should not be misinterpreted as a web service and it is not always required to be a WSDL description when exposing the service for a component. It is up to the vendor to implement the service.

Components and targets are wired together through binding, by which the component can communicate. Source and target components are wired to form the assembly. Bindings are the means by which the source and target communicate with each other through wires.

The following is a code snippet of `composite.xml`, where different components are wired together and services are binded:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Generated by Oracle SOA Modeler version 1.0 at [12/28/10 7:03
PM]. -->
<composite name="ProcessClaimsInfoApps" revision="1.0"
label="2010-12-28_19-03-39_049"
mode="active" state="on" xmlns="http://xmlns.oracle.
com/sca/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:orawsp="http://schemas.oracle.com/ws/2006/01/policy"
xmlns:ui="http://xmlns.oracle.com/soa/designer/">
  <import namespace="http://xmlns.oracle.com/MediatorTest/
ProcessClaimsInfoApps/ProcessClaims"
location="ProcessClaims.wsdl" importType="wsdl"/>
  <import namespace="http://xmlns.oracle.com/MediatorTest/
ProcessClaimsInfoApps/ReceiveClaimsInfo"
location="ReceiveClaimsInfo.wsdl" importType="wsdl"/>
  <import namespace="urn:GetEmployeeInformation"
location="GetEmployeeInformation.wsdl" importType="wsdl"/>
  <import namespace="urn:GetEmployeeInformation"
location="GetEmployeeInformationRef.wsdl"
importType="wsdl"/>
  <service name="processclaims_client_ep"
ui:wsdlLocation="ProcessClaims.wsdl">
    <interface.wsdl interface="http://xmlns.oracle.com/MediatorTest/
ProcessClaimsInfoApps/ProcessClaims#wsdl.interface(ProcessClaims)"/>
    <binding.ws port="http://xmlns.oracle.com/MediatorTest/
ProcessClaimsInfoApps/ProcessClaims#wsdl.endpoint(processclaims_
client_ep/ProcessClaims_pt)"/>
  </service>
  <service name="ReceiveClaimsInfo_ep" ui:wsdlLocation="ReceiveClaims
Info.wsdl">
    <interface.wsdl interface="http://xmlns.oracle.com/MediatorTest/
ProcessClaimsInfoApps/ReceiveClaimsInfo#wsdl.interface(execute_ptt)"
callbackInterface="http://xmlns.oracle.
com/MediatorTest/ProcessClaimsInfoApps/ReceiveClaimsInfo#wsdl.
interface(callback_ptt)"/>
    <binding.ws port="http://xmlns.oracle.com/
MediatorTest/ProcessClaimsInfoApps/ReceiveClaimsInfo#wsdl.
endpoint(ReceiveClaimsInfo_ep/execute_pt)"/>
    <callback>
      <binding.ws port="http://xmlns.oracle.com/
MediatorTest/ProcessClaimsInfoApps/ReceiveClaimsInfo#wsdl.
endpoint(ReceiveClaimsInfo_ep/callback_pt)"/>
    </callback>
  </service>
</component name="ProcessClaims">
```

---

```

    <implementation.bpel src="ProcessClaims.bpel"/>
  </component>
  <component name="ReceiveClaimsInfo">
    <implementation.mediator src="ReceiveClaimsInfo.mplan"/>
  </component>
  <reference name="GetEmployeeService"
    ui:wsdlLocation="GetEmployeeInformation.wsdl">
    <interface.wsdl interface="urn:GetEmployeeInformation#wsdl.
interface(getEmployeePort)"/>
    <binding.ws port="urn:GetEmployeeInformation#wsdl.
endpoint(getEmployeePort/getEmployeePortPort)"
    location="GetEmployeeInformation.wsdl"/>
  </reference>
  <wire>
    <source.uri>processclaims_client_ep</source.uri>
    <target.uri>ProcessClaims/processclaims_client</target.uri>
  </wire>
  <wire>
    <source.uri>ReceiveClaimsInfo_ep</source.uri>
    <target.uri>ReceiveClaimsInfo/ReceiveClaimsInfo</target.uri>
  </wire>
  <wire>
    <source.uri>ReceiveClaimsInfo/ProcessClaims.processclaims_client</
source.uri>
    <target.uri>ProcessClaims/processclaims_client</target.uri>
  </wire>
  <wire>
    <source.uri>ProcessClaims/GetEmployeeService</source.uri>
    <target.uri>GetEmployeeService</target.uri>
  </wire>
</composite>

```

---

Sr. no.	Elements in composite.xml	Description
1	Component	Components are executable units that process the business process logic
2	Service	Services are interface to composite that exposes the functionality provided by the SCA application
3	Bindings	Bindings are means by which exposed services are reached by the SCA runtime
4	Wire	Wires are the way by which the source and target components are integrated

---

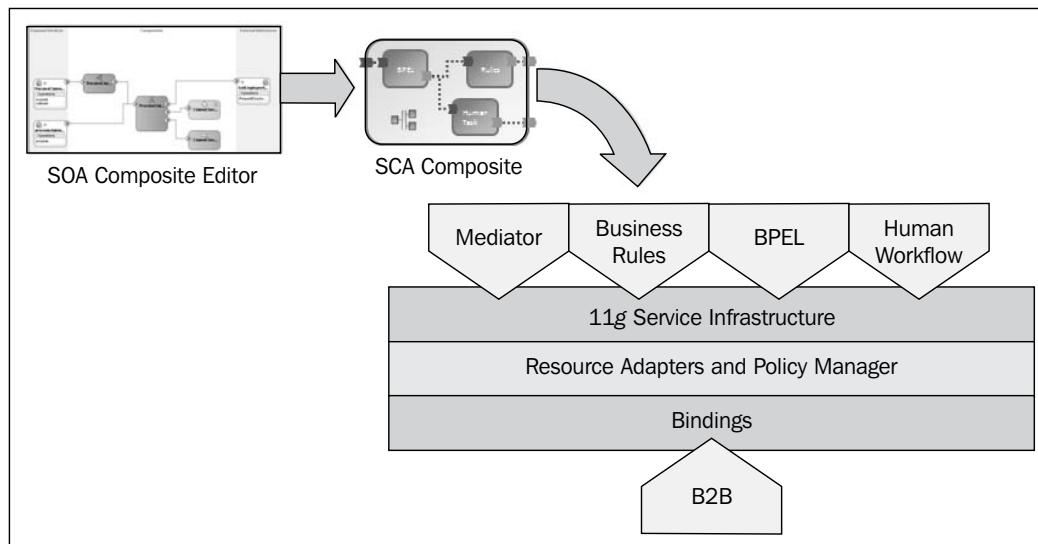


## SCA with Oracle SOA Suite 11g

Oracle SOA Suite 11g is built on the foundation of SCA specifications. Oracle SOA Suite 11g is re-architected from 10g in line with SCA specifications and a few extensions are implemented for the benefit of Oracle Fusion Middleware stack.

Oracle SOA Suite 11g has a unified runtime environment called 11g Service Infrastructure to execute various components of the Oracle Fusion Middleware stack. These components interact with each other using the 11g Service Infrastructure to exchange the processed data between them.

The following diagram shows the components involved in the SOA Suite 11g that enable SCA:



At design time, the composite editor is used to build the components and a deployable unit called composite is created to be deployed into 11g Service Infrastructure. Services exposed through SCA runtime are executed using the service engines available to execute the request provided by the client.

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Describing the SOA concepts:** We discussed in detail about the principle drivers of SOA such as interoperability, loose coupling, reusability, granularity, and versioning of services
- **Identifying standards that enable SOA:** We discussed in detail about the various SOA standards such as XML, XSD, XSLT, SOAP, MTOM, UDDI, XOP, WS-\*, and so on
- **Describing the Event Driven Architecture (EDA):** We understood about the significance of EDA with a real world example and features supported by Oracle SOA Suite 11g to enable them
- **Reviewing the Service Component Architecture (SCA):** We discussed in detail about the SCA specifications and features provided by Oracle SCA

## Self-review questions

1. Creating web service implementation from WSDL is a \_\_\_\_\_ approach in web service development.
  - a. Top-down
  - b. Bottom-up
  - c. None of the above
  - d. Web service development standards
2. XPath is a subset of XQuery.
  - a. True
  - b. False
3. Discovery of services are facilitated through:
  - a. WSDL
  - b. UDDI
  - c. SOA registry
  - d. REST

4. Of the following standards, which enables QOS in SOA design?
  - a. WS-Addressing
  - b. WS-I
  - c. WS-Reliable Messaging
  - d. WS-Security
5. XML can be represented in binary form.
  - a. True
  - b. False
6. Service component specification doesn't include:
  - a. SCA policy framework
  - b. SCA JMS binding
  - c. SCA JCA framework
  - d. None of the above
7. SCA supports implementation using all the programming languages except:
  - a. Java
  - b. PHP
  - c. C++
  - d. Groovy
8. Service data objects specifications are part of SCA specifications.
  - a. True
  - b. False
9. Auto-wiring feature is provided as a part of SCA specification.
  - a. True
  - b. False
10. Business events are currently part of SCA specifications.
  - a. True
  - b. False

11. Event definitions are stored in \_\_\_\_\_.
- a. EDL
  - b. MDS
  - c. XML
  - d. Oracle SOA database
12. SCA has a complementary standard for Oracle EDN.
- a. True
  - b. False



# 3

## SOA Composite Application

Composite Application is the next new trend in software development; though the inspiration for this model of application development has long been known to us, the methodology adopted to develop this type of application is new to the software development community. Composing the service together to build an application is a new paradigm in software development which provides greater flexibility and manageability to build and integrate applications.

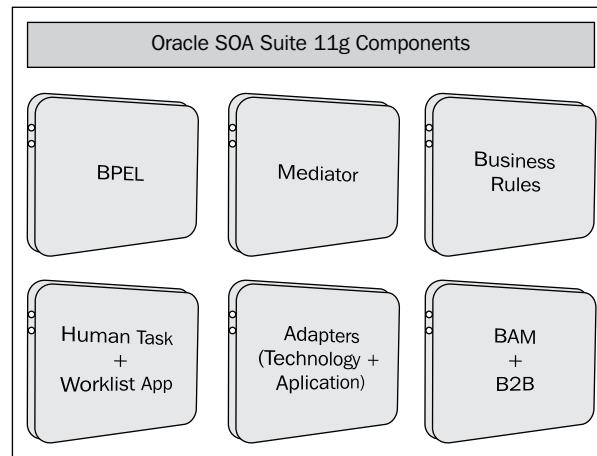
Oracle SOA Composite Applications are built upon the SCA Architecture. This architecture will facilitate composing application components together to build the integration components or applications that can be accessed as services.

This chapter will discuss in detail the following exam objectives:

- Describe Oracle SOA Suite 11g components
- Describe the Service Components
- Define a composite application
- Describe the SOA Composite Editor

## Oracle SOA Suite 11g components

Oracle SOA 11g Components comprise three categories – Service Components, Adapters, and Bindings.



## Oracle SOA component description: BPEL

BPEL expands to **Business Process Execution Language**. BPEL is an essential tool in the SOA world to achieve the Business Process Orchestration. It is a very thin and lightweight XML-based language that can be used to accomplish the Process Orchestration. BPEL uses web services exposed from different business applications and information systems (through technology adapters).

OASIS drafted BPEL standards by working closely with various industry's leading companies such as IBM, Oracle, Amber Point, SAP, SOA Systems, iWay Software, and so on. It is predicted that BPEL will be the next widely used language in most enterprises after Java.

In Oracle SOA Suite 11g, Oracle BPEL Process Manager is integrated to Oracle SCA Runtime. This helps the Oracle SCA runtime to integrate the various Oracle SOA components together to accomplish some very interesting tasks. Oracle BPEL Process Manager, introduced in 10g, is now integrated into Oracle SCA runtime components as a process engine along with Mediator, Business Rule, and Human Workflow Engines.

## **Oracle SOA component description: Mediator**

Oracle Mediator provides a lightweight framework to mediate data between various components within a composite application. Mediator converts data to facilitate communication between different interfaces exposed by different components, which are wired together to build an SOA composite application. A Mediator component can be used to consume a business event or to receive a service invocation. A Mediator component can evaluate routing rules, perform transformations, validate, and either invoke another service or raise another business event.

For example, a Mediator can accept data contained in a text file from an application or service, transform it to a format appropriate for updating a database that serves as a customer repository, and then route and deliver the data to that database. Oracle Mediator facilitates integration between events and services, where service invocations and events can be mixed and matched.

## **Oracle SOA component description: Business Rules**

Rules and policies are part of all business processes defined in different zones of business. Defining the business rules and policies helps in effective execution of business and it creates a reliable business environment. Business Rules are defined by the organization as well as by governments to conduct business in an ethical and moral way. Business Rules are also created to be more competitive as well as identify the right customers and serve them with more care. The following are some examples of Business Rules defined in different zones of a business environment:

- If the total annual income of the family is greater than \$80,000, then auto approve the loan, as it is less than \$120K, with 3 percent interest
- If the age of a customer is younger than 18, then decline sales of products that are classified under Section 18 of Govt. Act 1937 to that person
- If traveler account has total miles for the year greater than 500K, then change the customer status to platinum
- If the customer payment history is excellent and rated as gold, provide 5 percent additional discount on the new services subscribed by the customer for the next nine months



## Oracle SOA component description: Human Task

Oracle Human Workflow helps to accomplish automated notification on the process movement or transition between different phases of business and provides the necessary frameworks and tools to ensure the right checks and balances in the business process. Wider adoption of SOA requires centralized workflow management tools. Oracle SOA Suite provides this capability to extend the core business process execution (orchestration) framework to include the human component in the business process automation. Though workflow has been a well known technology for many years and adopted widely in different business applications like Oracle E-Business Suite, PeopleSoft, Siebel, and so on, changes in the fundamental approach to Fusion Application and SOA, call for centralized workflow management tool. Oracle Human Workflow Architecture provides an extensive list of features, which are as follows:

- Ability for human beings to interact with business processes that include assignment and routing of tasks to the right users or user group
- Ability to generate notifications related to deadlines, escalations, and other generic normal notifications that would aid in timely execution of the tasks associated with the business process
- Ability to view the task online and provide approvals or reject the request generated through business process automation
- Ability to organize, filter, and prioritize the tasks of the end users to productively perform their task
- Ability to manage the vacation and other features required by supervisors and business owners to manage the performance of tasks

## Oracle SOA component description: Adapter

Adapters are the heart of any middleware suite. Without adapters, middleware components would not be able to perform the task of enterprise application integration.

Oracle SOA Suite 11g has a very extensive list of adapters that can be used in technology and application integration. Oracle adapters follow the JCA (J2EE Connector Architecture) framework. Oracle delivers adapters in the following categories:

- Oracle technology adapters
- Packaged-application adapters
- Legacy adapters
- Adapters for Oracle applications

The following table describes each adapter in detail:

Sr.No	Component Type	Component Name	Description
1	Adapter	ADF-BC Service	Adapter used to connect to ADF-BC components using SDO objects
2	Adapter	AQ Adapter	Adapter used to connect to Oracle AQ
3	Adapter	Database Adapter	Adapter used to connect to the database
4	Adapter	EJB Service	Adapter used to connect to the EJB service
5	Adapter	File Adapter	Adapter used to connect to a file
6	Adapter	FTP Adapter	Adapter used to connect to an FTP site
7	Adapter	JMS Adapter	Adapter used to connect to a JMS server
8	Adapter	MQ Adapter	Adapter used to connect to an MQ server
9	Adapter	Socket Adapter	Adapter used to connect to TCP/IP
10	Adapter	Web Service	Adapter used to connect to web service
11	Adapter	B2B Adapter	Adapters used to connect to the Oracle B2B Application to convert SOA Messages to B2B Messages
12	Adapter	Oracle Applications	Adapter used to connect to an Oracle Application
13	Adapter	BAM Adapter	Adapter used to connect to Business Analysis Modeler

## Oracle SOA component description: Business Activity Monitor

Oracle Business Activity Monitoring (BAM) enables us to monitor business services and processes in an enterprise, correlate key performance indicators (KPIs), and change business processes, or take corrective actions if the business environment changes.

## Oracle SOA component description: B2B

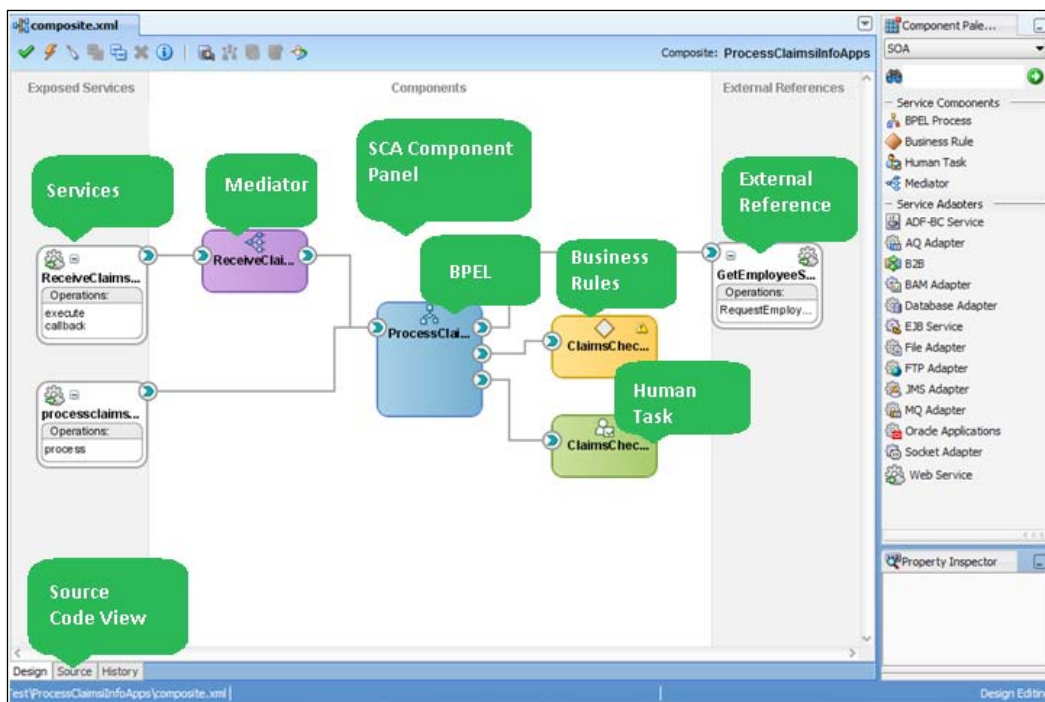
Oracle B2B is a gateway that enables secure and reliable exchange of business documents between an enterprise and its trading partners. Oracle B2B supports business-to-business document standards, security, transports, messaging services, and trading partner management. Oracle B2B is a binding component that enables the implementation of e-commerce business processes. Oracle B2B supports transformation and transfer of EDI messages between different trading partners.

## Oracle SOA component description: MDS

Oracle MDS—Meta Data Store is used to store metadata for Oracle Fusion middleware components and composite application state information.

## SOA Composite Editor

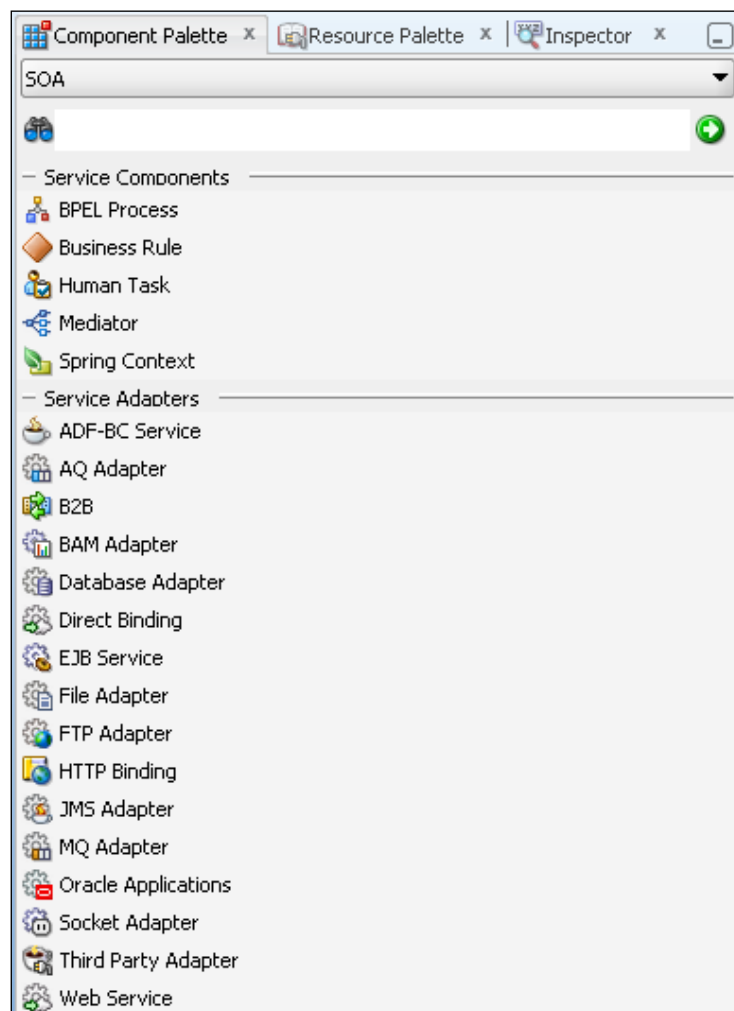
SOA Composite Editor exposes the services through exposed service links and components' swim-lane is used to place the components that are part of the composite application. **Component Palette** contains the list of components to be used in the SCA application.



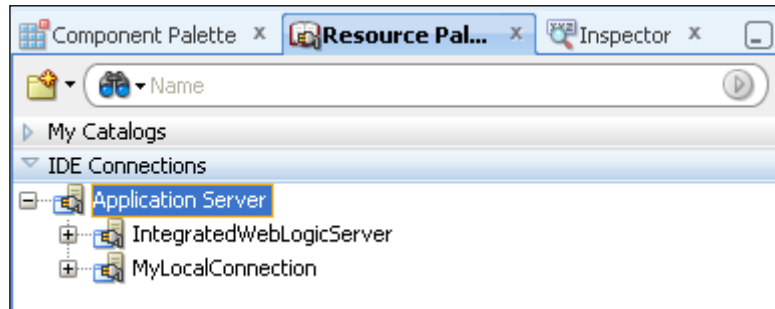
**Composite Editor** provides the following features:

- Ability to wire and bind service components
- Ability to configure WS-Policies
- Ability to generate Deployment Configuration Plans
- Ability to configure wire adapters with service components
- Ability to configure adapters through Adapter Configuration Wizards

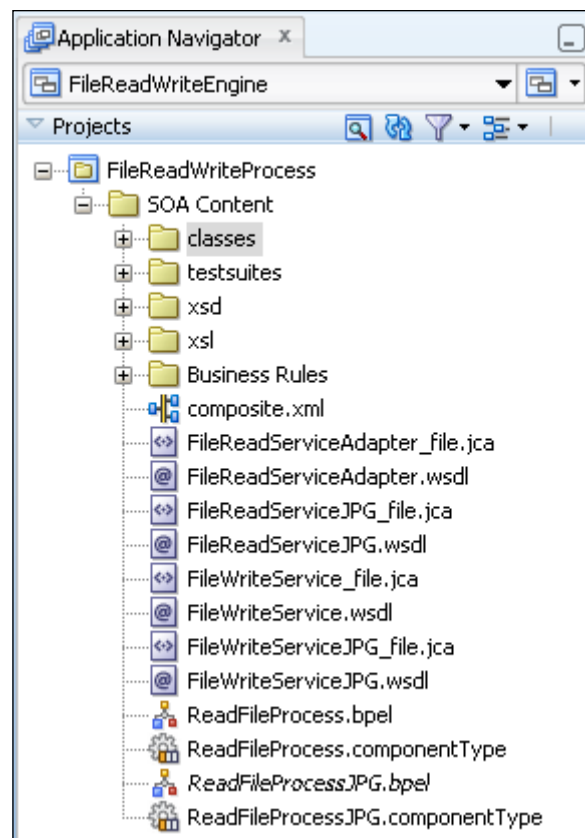
**Component Palette** of the SOA Composite view is shown in the following screenshot. This palette shows the list of **Service Components** and **Service Adapters**.



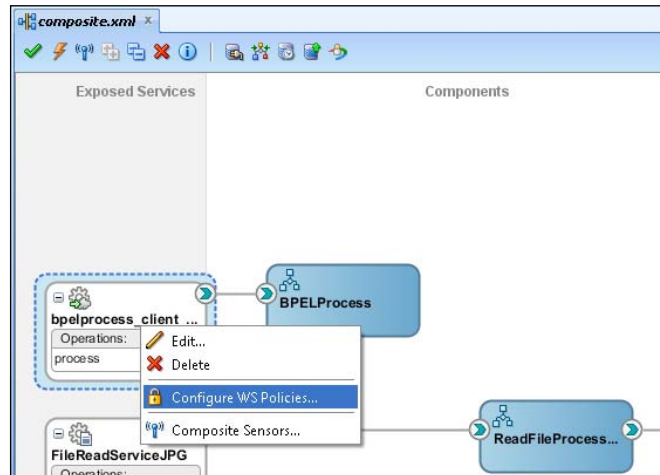
**Resource Palette** shows the application-specific connections and so on:



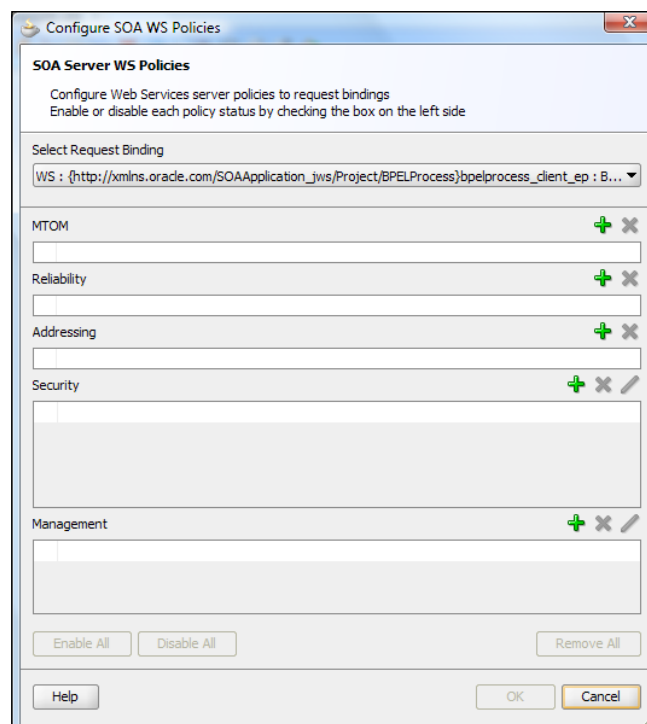
The **Application** and **Project Explorer** view is shown in the following screenshot:



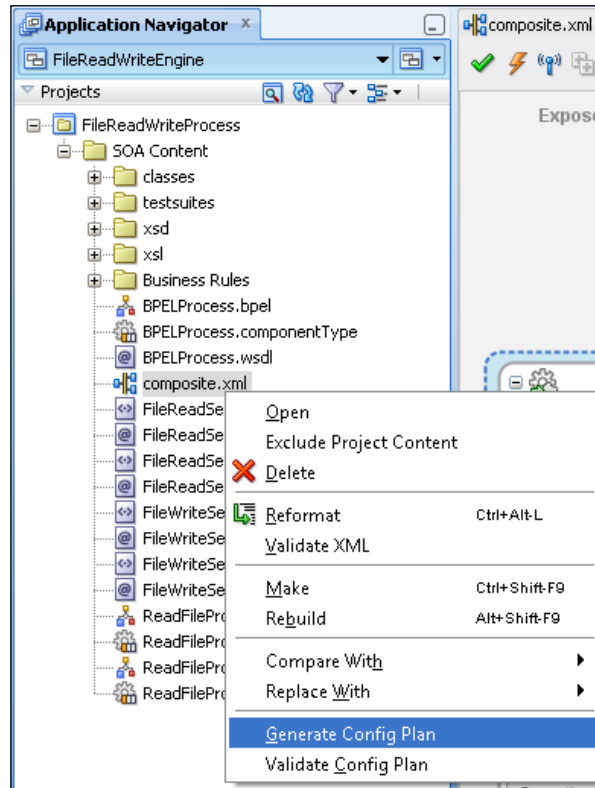
WS Policies for the SOA Composite are configured using the composite editor:



SOA WS Policies dialog opens to configure required policies under the **MTOM**, **Reliability**, **Addressing**, **Security**, and **Management** sections using the required policies.



The SOA Composite Editor can be used to generate a deployment configuration plan:



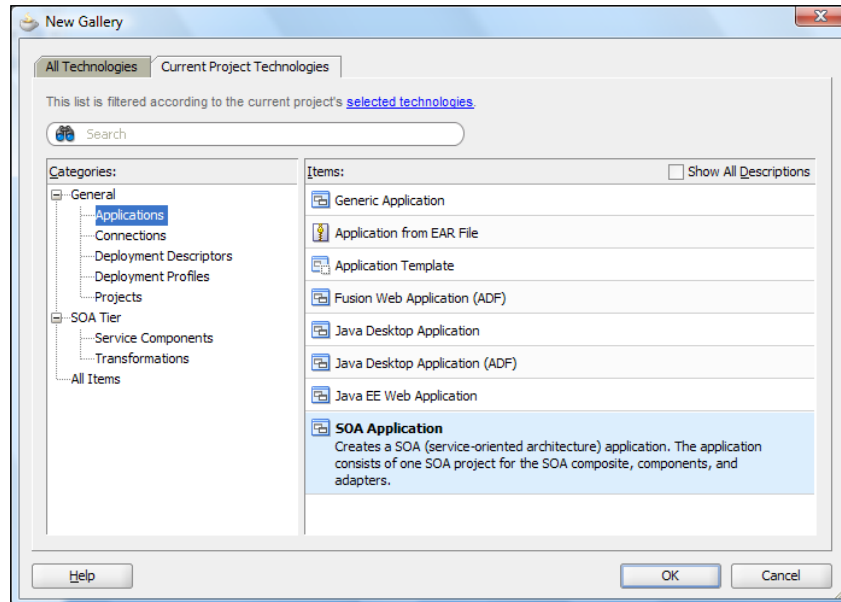
## Defining an Oracle SOA Composite application

An Oracle SOA Composite application is defined using the SOA application category from JDeveloper. The following are the steps involved in creating the Oracle SOA Composite application:

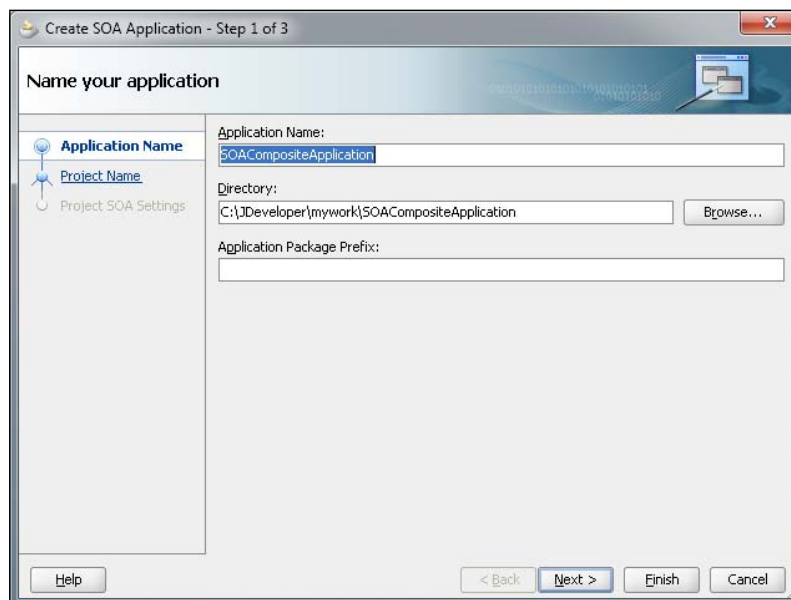
1. Select **File | New** from JDeveloper.



2. Choose the **Applications** from categories and **SOA Application** as an item:

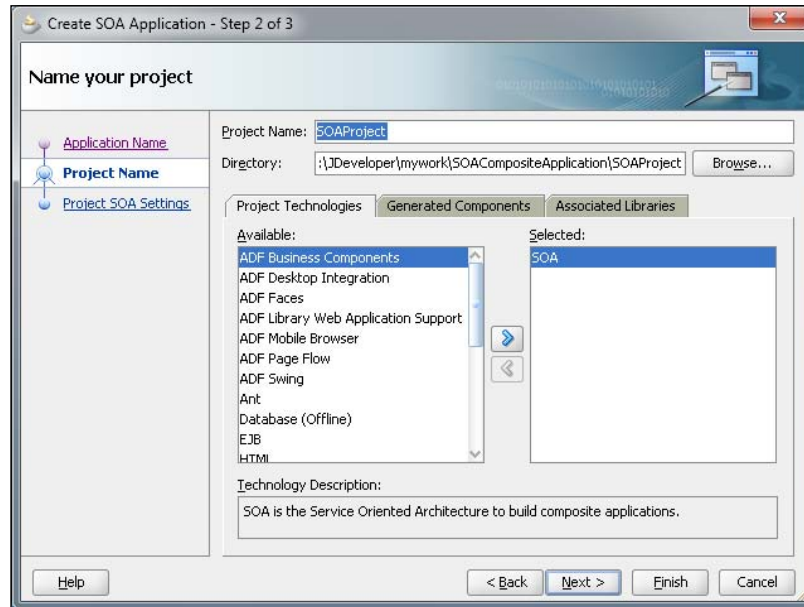


3. Provide the **Application Name** as SOACompositeApplication and click on **Next**:

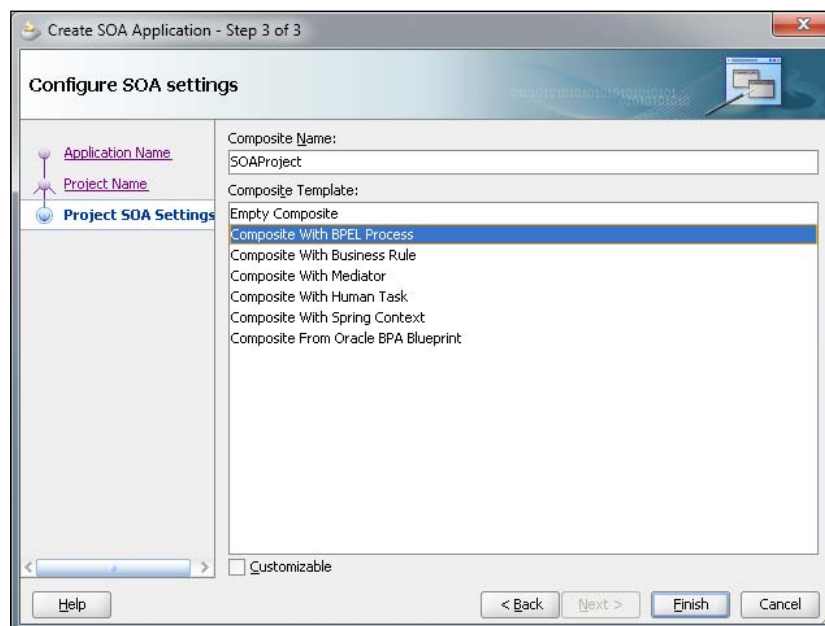




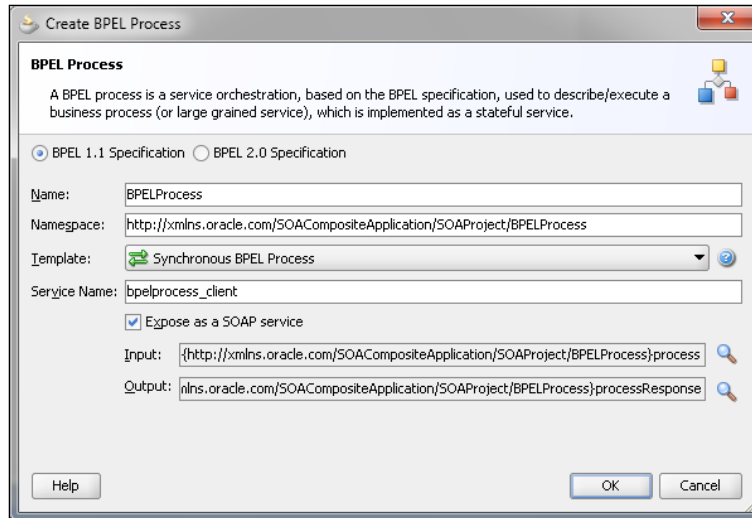
4. Provide the **Project Name** as SOAPProject and click on **Next**:



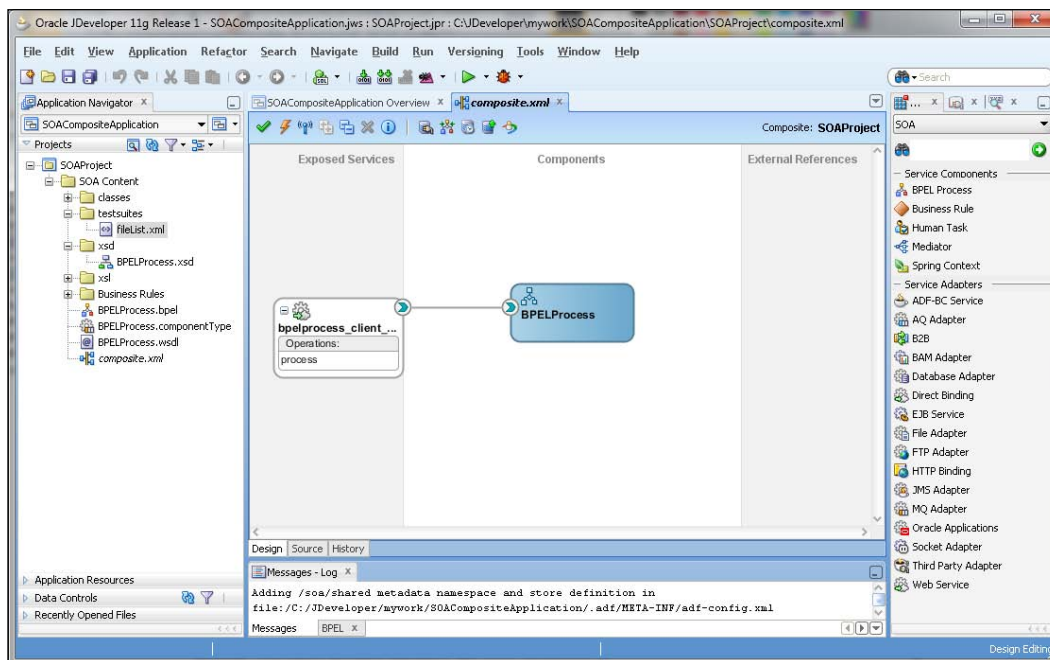
5. The Oracle SOA Composite application is created using the Wizard and following the steps shown:



6. Provide **BPEL Name** as `BPELProcess` and select the **Template** as **Synchronous BPEL Process**:



7. The Composite application is created using the Composite Editor:



The following is a list of artifacts that are created when the Oracle SOA Composite application is defined:

S.No	Component Type	Filename	Description
1	SCA Composite	Composite.xml	This contains the description of the SOA Composite application
2	For every SOA Component	*.componentType	This file contains the details of the services and references the component is referring to
3	BPEL	*.bpel	This file contains orchestration details about the service component being used
4	Business Rule	*.decs	Description about the Business Rule
5	Business Rule Dictionary	*.rules	Rules dictionary
6	Component WSDL	*.wsdl	Service description of the Component being exposed as a service
7	Human Workflow Task	*.task	Task description for Human Workflow
8	Mediator Plan	*.mplan	Mediation details about the SCA layer are available here
9	Adapters	*.jca	This file contains the connection factory related information to connect to the underlying EIS system
10	Adapters	*.wsdl	This is the WSDL that describes the service operation executed by WSDL

The following is a sample Composite.xml with a Mediator, BPEL, Business Rule, Human Task with a File, and FTP Adapter:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Generated by Oracle SOA Modeler version 1.0 at [1/22/11 5:38 PM].
-->
<composite name="OracleSOAProject" revision="1.0"
label="2011-01-22_17-38-16_955" mode="active"
state="on"xmlns="http://xmlns.oracle.com/sca/1.0"xmlns:xs=http://
www.w3.org/2001/XMLSchemaxmlns:wsp="http://schemas.xmlsoap.org/
ws/2004/09/policy xmlns:orawsp=http://schemas.oracle.com/ws/2006/01/
policyxmlns:ui="http://xmlns.oracle.com/soa/designer/">
<import namespace="http://xmlns.oracle.com/OracleSOAApplication/
OracleSOAProject/OracleSOABPELComponent"
```

---

```

        location="OracleSOABPELComponent.wsdl"importType="wsdl"/>
<import namespace="http://xmlns.oracle.com/OracleSOAApplication/
OracleSOAProject/OracleSOAMediator"
        location="OracleSOAMediator.wsdl"importType="wsdl"/>
<import namespace="http://xmlns.oracle.com/OracleSOABusinessRules/
OracleSOABusinessRules_DecisionService"
        location="OracleSOABusinessRules_DecisionService.wsdl"
importType="wsdl"/>
<import namespace="http://xmlns.oracle.com/bpel/workflow/taskService"
        location="oramds:/soa/shared/workflow/TaskServiceInterface.
wsdl"
importType="wsdl"/>
<import namespace="http://xmlns.oracle.com/pcbpel/adapter/file/
OracleSOAApplication/OracleSOAProject/FileAdapter%2F"
        location="FileAdapter.wsdl"importType="wsdl"/>
<import namespace="http://xmlns.oracle.com/pcbpel/adapter/ftp/
OracleSOAApplication/OracleSOAProject/FTPAdapter%2F"
        location="FTPAdapter.wsdl"importType="wsdl"/>
<service name="OracleSOAMediator_ep"ui:wsdlLocation="OracleSOAMediat
or.wsdl">
<interface.wsdl interface="http://xmlns.oracle.com/
OracleSOAApplication/OracleSOAProject/OracleSOAMediator#wsdl.
interface(execute_ptt)"/>
<binding.ws port="http://xmlns.oracle.com/OracleSOAApplication/
OracleSOAProject/OracleSOAMediator#wsdl.endpoint(OracleSOAMediator_ep/
execute_pt)"/>
</service>
<service name="OracleSOABusinessRules_DecisionService_ep"
ui:wsdlLocation="OracleSOABusinessRules_DecisionService.wsdl">
<interface.wsdl interface="http://xmlns.oracle.com/
OracleSOABusinessRules/OracleSOABusinessRules_DecisionService#wsdl.
interface(IDecisionService)"/>
<binding.ws port="http://xmlns.oracle.com/OracleSOABusinessRules/
OracleSOABusinessRules_DecisionService#wsdl.
endpoint(OracleSOABusinessRules_DecisionService_ep/IDecisionService_
pt)"/>
</service>
<component name="OracleSOABPELComponent">
<implementation.bpelsrc="OracleSOABPELComponent.bpel"/>
</component>
<component name="OracleSOAMediator">
<implementation.mediatorsrc="OracleSOAMediator.mplan"/>
</component>
<component name="OracleSOABusinessRules">
<implementation.decisionsrc="OracleSOABusinessRules.decs"/>
</component>

```

```
<component name="OracleSOAHumanTask">
<implementation.workflowsrc="OracleSOAHumanTask.task"/>
</component>
<reference name="FileAdapter"ui:wsdlLocation="FileAdapter.wsdl">
<interface.wsdl interface="http://xmlns.oracle.com/pcbpel/adapter/
file/OracleSOAApplication/OracleSOAPProject/FileAdapter%2F#wsdl.
interface(Write_ptt)"/>
<binding.jcaconfig="FileAdapter_file.jca"/>
</reference>
<reference name="FTPAdapter"ui:wsdlLocation="FTPAdapter.wsdl">
<interface.wsdl interface="http://xmlns.oracle.com/pcbpel/adapter/
ftp/OracleSOAApplication/OracleSOAPProject/FTPAdapter%2F#wsdl.
interface(Put_ptt)"/>
<binding.jcaconfig="FTPAdapter_ftp.jca"/>
</reference>
<wire>
<source.uri>OracleSOAMediator_ep</source.uri>
<target.uri>OracleSOAMediator/OracleSOAMediator</target.uri>
</wire>
<wire>
<source.uri>OracleSOAMediator/OracleSOABPELComponent.
oraclesoabpelcomponent_client</source.uri>
<target.uri>OracleSOABPELComponent/oraclesoabpelcomponent_client</
target.uri>
</wire>
<wire>
<source.uri>OracleSOABusinessRules_DecisionService_ep</source.uri>
<target.uri>OracleSOABusinessRules/OracleSOABusinessRules_
DecisionService</target.uri>
</wire>
<wire>
<source.uri>OracleSOAMediator/OracleSOABusinessRules.
OracleSOABusinessRules_DecisionService</source.uri>
<target.uri>OracleSOABusinessRules/OracleSOABusinessRules_
DecisionService</target.uri>
</wire>
<wire>
<source.uri>OracleSOABPELComponent/OracleSOAHumanTask.TaskService</
source.uri>
<target.uri>OracleSOAHumanTask/TaskService</target.uri>
</wire>
</composite>
```

Sample Component type description—the following file describes the services and references used by the component:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Generated by Oracle SOA Modeler version 1.0 at [1/22/11 5:39 PM].
-->
<componentType
xmlns="http://xmlns.oracle.com/sca/1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:ui="http://xmlns.oracle.com/soa/designer/">
  <service name="oraclesoabpelcomponent_client"
ui:wsdlLocation="OracleSOABPELComponent.wsdl">
    <interface.wsdl interface="http://xmlns.oracle.com/
OracleSOAApplication/OracleSOAPProject/OracleSOABPELComponent#wsdl.inte
rface(OracleSOABPELComponent)"/>
  </service>
  <reference name="OracleSOAHumanTask.TaskService"
ui:wsdlLocation="oramds:/soa/shared/workflow/TaskServiceInterface.
wsdl">
    <interface.wsdl interface="http://xmlns.oracle.com/bpel/workflow/
taskService#wsdl.interface(TaskService)"
        callbackInterface="http://xmlns.oracle.com/bpel/
workflow/taskService#wsdl.interface(TaskServiceCallback)"/>
  </reference>
</componentType>
```

A Sample Business Rule Decision service description is provided in the following file:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<decisionServicesxmlns="http://xmlns.oracle.com/bpel/rules"
name="OracleSOABusinessRules">
  <ruleEngineProvider name="OracleRulesSDK"
provider="Oracle_11.0.0.0.0">
    <repository type="SCA-Archive">
      <path>OracleSOAPProject/oracle/rules/oraclesoaproject/
OracleSOABusinessRules.rules</path>
    </repository>
  </ruleEngineProvider>
  <decisionService targetNamespace="http://xmlns.oracle.com/
OracleSOABusinessRules/OracleSOABusinessRules_DecisionService"ruleEn
gineProviderReference="OracleRulesSDK" name="OracleSOABusinessRules_
DecisionService">
    <catalog>OracleSOABusinessRules</catalog>
    <pattern name="CallFunctionStateless">
      <arguments>
        <call>oraclesoaproject.OracleSOABusinessRules.OracleSOABusinessRules_
DecisionService</call>
```

```
</arguments>
</pattern>
<pattern name="CallFunctionStateful">
  <arguments>
    <call>oraclesoaproject.OracleSOABusinessRules.OracleSOABusinessRules_
    DecisionService</call>
  </arguments>
</pattern>
</decisionService>
</decisionServices>
```

The following is a sample JCA file:

```
<adapter-config name="FTPAdapter" adapter="Ftp Adapter" xmlns="http://
platform.integration.oracle/blocks/adapter/fw/metadata">

  <connection-factory location="eis/Ftp/FtpAdapter" adapterRef="" />
  <endpoint-interaction portType="Put_ptt" operation="Put">
    <interaction-spec className="oracle.tip.adapter.ftp.outbound.
    FTPInteractionSpec">
      <property name="FileType" value="binary"/>
      <property name="PhysicalDirectory" value="/oracle"/>
      <property name="FileNamingConvention" value="PO_%SEQ%.txt"/>
      <property name="Append" value="false"/>
      <property name="NumberMessages" value="1"/>
    </interaction-spec>
  </endpoint-interaction>

</adapter-config>
```

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Describe Oracle SOA Suite 11g components and service components:**  
We discussed in detail about the various Oracle SOA Suite 11g components such as BPEL, Mediator, Business Rules, Adapters, Human Workflow, BAM, and B2B
- **Describe the SOA Composite Editor:** We discussed about the various panes that are available to JDeveloper to define the SOA Composite
- **Define a composite application:** We understood the process in defining an Oracle SOA Composite application and associated steps and actions

## Self-review questions

1. Business Rule is the component used to:
  - a. Route Services in a Composite
  - b. Orchestrate Business Process flows
  - c. Monitor and Analyze the Business Process changes
  - d. Define conditions that will change the business process flow or outcome/path of the business process
2. Which of the following is not a Service Component of an Oracle SOA Composite application?
  - a. BPEL
  - b. Mediator
  - c. Human Task
  - d. File Adapter
3. Which of the following components is used to execute or orchestrate a business process?
  - a. BPEL
  - b. Mediator
  - c. CEP Engine
  - d. None of the above
4. A Composite application is defined through
  - a. `bep1.xml`
  - b. `*.mplan`
  - c. `composite.xml`
  - d. `build.xml`
5. B2B is a reference component in Oracle SOA 11g Infrastructure.
  - a. True
  - b. False
6. SOA Fault Policy can be specified from the SOA Composite Editor.
  - a. True
  - b. False



7. Oracle SOA Components can be tested from JDeveloper.
  - a. True
  - b. False
8. WS-Policies can be attached to the Component from Oracle SOA Design Time.
  - a. True
  - b. False
9. DB Connection to MDS is required to create an MDS connection.
  - a. True
  - b. False
10. SCA is the Oracle Proprietary standard.
  - a. True
  - b. False

# 4

## Working with Adapters

Adapters are the heart of any middleware suite. Without adapters, middleware components would not be able to perform the task of Enterprise Application Integration.

Oracle SOA Suite 11g has an extensive list of adapters that can be used in technology and application integration. Oracle adapters follow the JCA (J2EE Connector Architecture) framework. Oracle delivers adapters in the following categories:

- Oracle technology adapters
- Packaged-application adapters
- Legacy adapters
- Adapters for Oracle applications

This chapter will guide you to understand the various adapters that are available with Oracle SOA Suite 11g and prepare you to meet the following exam objectives:

- Describe the adapter concepts and framework
- Describe Technology adapters: File, Database, JMS, and so on
- Describe Applications adapters: E-Biz Suite, PeopleSoft, Siebel, and so on
- Explain adapter design-time configuration
- Explain adapter runtime configuration

## Introduction to Java Connector architecture

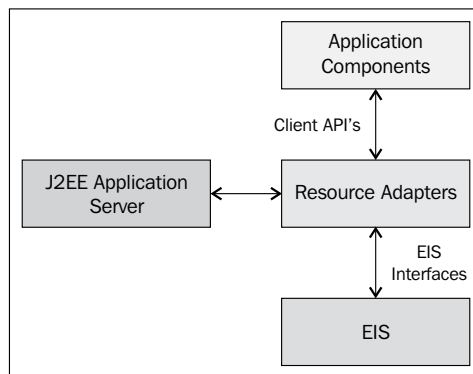
This section of the chapter will explain the concepts of JCA (J2EE Connector Architecture) before we dive deep into Oracle adapters.

JCA was developed as part of the J2EE standards to promote open standards across the Enterprise Application Integration domain. JCA promotes development of adapters for various Enterprise Information Systems.

JCA specification, developed as part of JSR 322, abstracts the system level implementation for integrating J2EE applications with Enterprise Information System.

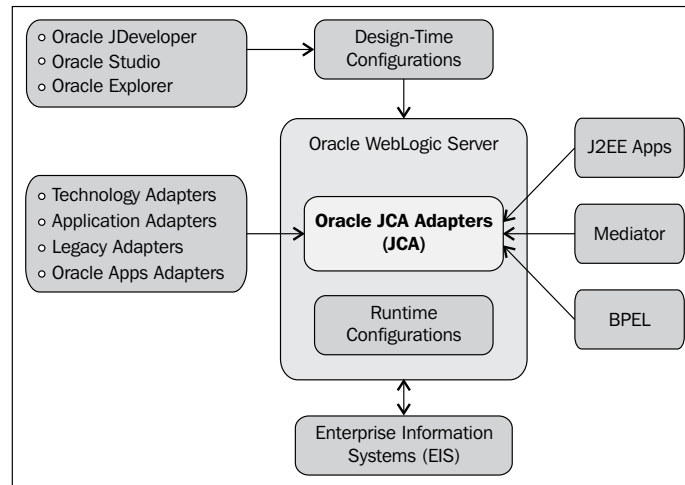
JCA helps to achieve abstraction in the following:

- Connection management with EIS
- Transaction management with EIS
- Establish the security for EIS connectivity
- Life cycle management of resource adapters
- Work management for resource adapters with the application server
- Transaction in-flow management into the application server
- Message inflow management within the application server



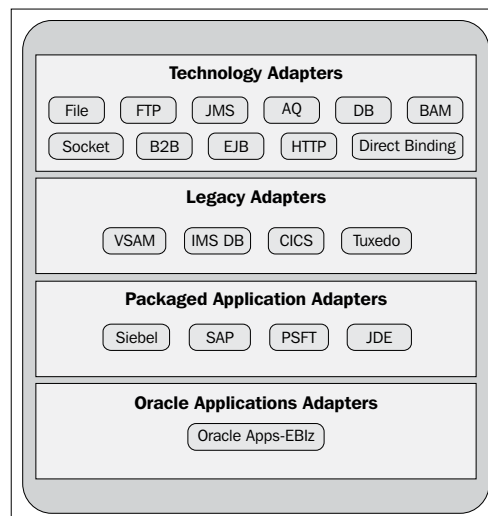
## Adapter concepts and framework

The following diagram depicts the Oracle Adapter framework:



JDeveloper, Oracle Studio, and Application Explorer are used to build and create the design-time configurations of the Oracle adapter framework.

Oracle SOA Suite uses Technology Adapters to connect to technology platforms such as file, FTP, MQ, AQ, JMS-based queues, and so on. Oracle Application Adapters such as PeopleSoft, Siebel, JD Edwards, SAP, Salesforce.com. Refer to the following diagram for more details are used to connect the packaged application:



Oracle JDeveloper provides the features to allow the integration developers to configure the adapter at design time and customize the configurations for various different environments through the deployment configuration framework (configuration plans).

Oracle Fusion Middleware integration server integrates Resource adapters from design time through JCA binding details. The runtime component of JCA framework aids in converting the XML message to J2CA interaction to accomplish the required level of interactions/integration with Enterprise Information Systems.

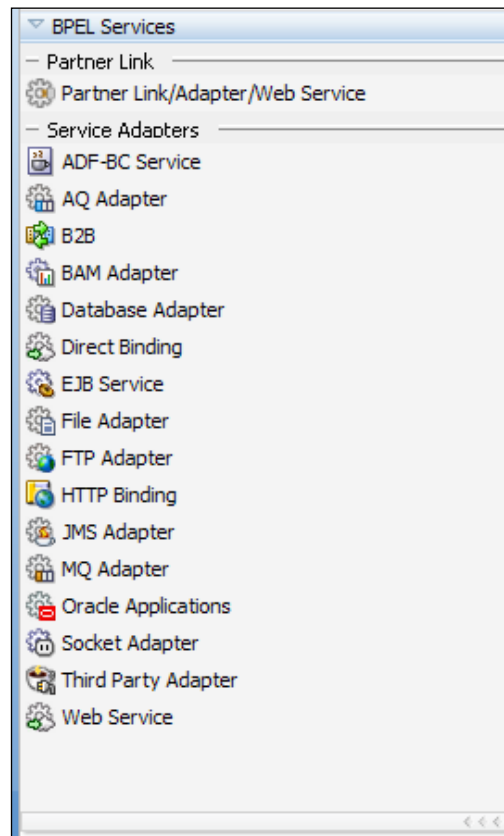
Oracle SOA Suite has the required features to integrate the Oracle SCA Components (BPEL and Mediator) to accomplish the enterprise integration objectives for process and data integrations. The transaction management capabilities of Oracle BPEL Process Manager, Mediator Engine, WebLogic application server, and JCA framework will aid in the right level of transaction management for the integrations.

Oracle provides the following list of design-time environment for specific adapters:

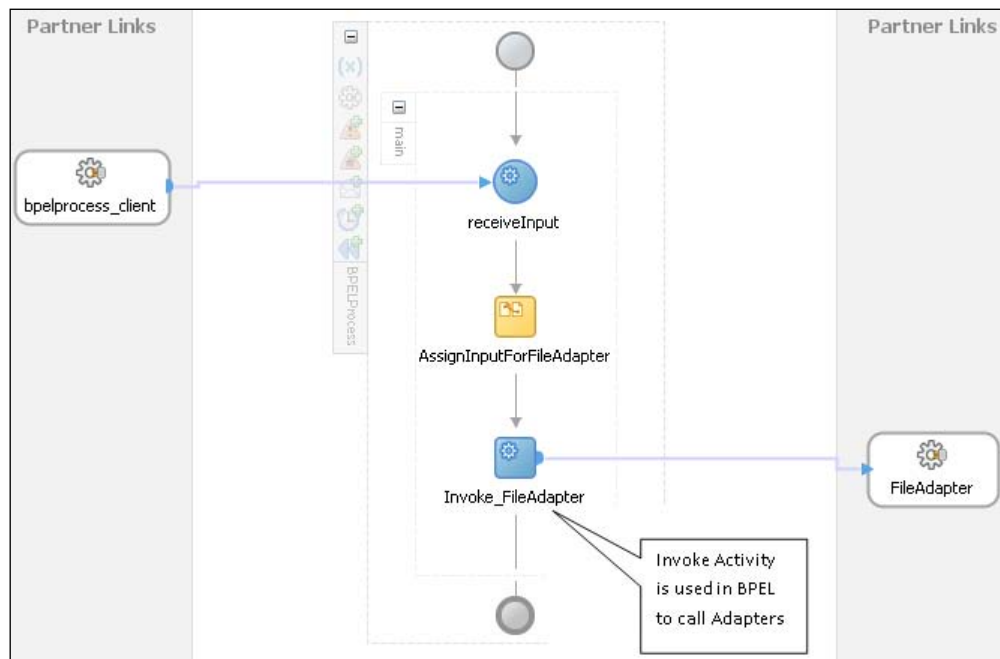
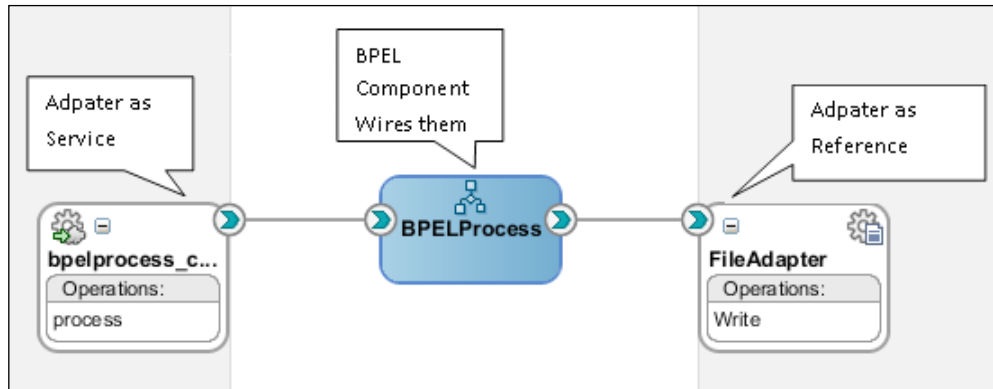
S.No	Design Time Tool	Adapters
1	JDeveloper	Technology adapters (File, FTP, MQ, AQ, Socket)
2	Oracle Studio	Legacy adapters (VSAM, IMS DB, CICS, Tuxedo)
3	Adapter Application Explorer	Packaged application adapters (Siebel, SAP, PeopleSoft and JD Edwards)

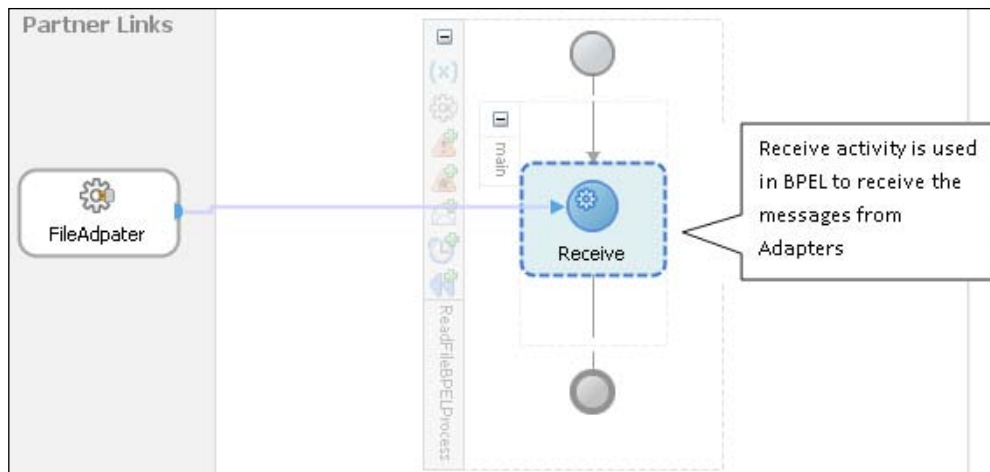
## Technology adapters: File, Database, JMS, and so on

The following screenshot shows the service adapters that are available in Oracle SOA Suite 11g (11.1.1.5):



This pane allows the adapters to be selected and configured in the swim-lane of SCA components. Adapters can be configured both as a service as well as a reference. Based on the operation that is selected in the adapter configuration, the adapter can be configured as a service or a reference.





S.No	Name of the Adapter	Description
1	File Adapter	<p>File Adapter is an extensively used adapter that will help in reading, writing, and listing folder contents. This adapter has the special feature of reading the file metadata information without reading the contents of the file. File Adapter used Activation Specifications for Inbound Actions (Service) and Interaction Specification for Outbound Actions (Reference). File Adapter provides various interesting features such as include pattern and exclude pattern.</p> <p>File and FTP Adapter provides a feature to poll a directory or an FTP site based on the trigger file.</p> <p>Note: File Adapter does not require any runtime configuration. Design-time configurations are sufficient to execute the instructions configured for this adapter.</p>
2	FTP Adapter	<p>FTP Adapter is used extensively to put, get, and list the file contents, as provided by File Adapter. In 11g, FTP Adapter has the feature to read the contents of the file recursively in the subdirectories as well. Oracle FTP Adapter also supports features to access SFTP sites. However, the FTP Adapter doesn't provide a feature to use RESTART AND RECOVERY options during a large file transfer.</p> <p>Note: Additional Runtime Configuration is applicable for this adapter.</p>



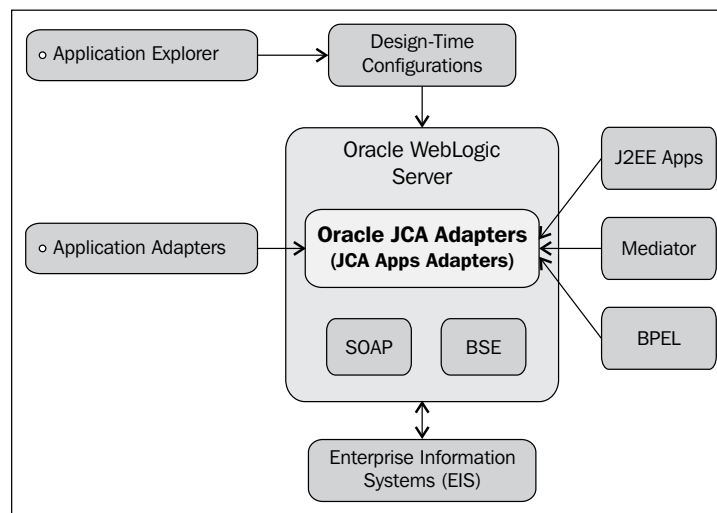
S.No	Name of the Adapter	Description
3	Database Adapter	<p>Database Adapter provides features to connect the SQL databases and JDBC providers. Database Adapter provides features to call a SQL procedure, perform standard SQL operations, such as SELECT, INSERT, UPDATE, DELETE, MERGE, poll for new record, and execute a native SQL from Fusion Middleware Components.</p> <p>Note: Database Adapter requires JNDI entries; it should be configured in the application server to resolve the right database against with the configured action to be performed.</p>
4	JMS Adapter	<p>JMS Adapter helps to connect any messaging system that adheres to JMS specifications. Currently, Oracle has certified the adapter to work with AQ JMS, TIBCO JMS, IBM WebSphere MQ Series, WebLogic JMS, Apache, and Active MQ.</p> <p>Oracle JMS Adapter supports the integration with both topics and queues.</p>
5	MQ Adapter	<p>MQ Adapter is used to expose the features of IBM MQ Series.</p>
6	AQ Adapter	<p>Oracle AQ Adapter provides the features to connect to Oracle AQ Messaging System. It allows interacting with both single consumer and multi consumer queue. Oracle AQ Adapter confusing.</p> <p>Oracle AQ Adapter provides the following enqueue features:</p> <ul style="list-style-type: none"><li>• Multi consumer queue</li><li>• Message priority</li><li>• Time specifications and scheduling</li><li>• Correlation identifier</li></ul> <p>Oracle AQ Adapter provides the following dequeue features:</p> <ul style="list-style-type: none"><li>• Poll option</li><li>• Notification option</li></ul>

S.No	Name of the Adapter	Description
		<p>Oracle AQ Adapter also supports the following features as well:</p> <ul style="list-style-type: none"> <li>• Retries with delays</li> <li>• Rule-based subscriptions</li> <li>• Oracle adapter header properties</li> <li>• Dequeue condition</li> </ul>
7	Socket Adapter	Socket Adapter provides a feature to interact with standard and nonstandard protocols for communicating via TCP/IP Sockets.
8	BAM Adapter	BAM Adapter provides the feature to interact with Oracle BAM components from the SCA environment.
9	B2B Adapter	B2B Adapter provides the feature to interact with Oracle B2B gateway product to provide seamless integration between Oracle SOA Suite and Oracle B2B components.
10	ADF-BC Adapter Service	The Oracle ADF business component service is used for connecting Oracle ADF applications using Service Data Object (SDO). Service Data Objects simplify the representation of associated data in SOA composite applications.
11	EJB Service Adapter	EJB Service Adapter is used to integrate Enterprise JavaBeans with SOA composite applications. Integration is achieved through the use of service data object (SDO) parameters.
12	Direct Binding Adapter	Direct Binding Adapter is used to integrate with Oracle Service Bus or Oracle SOA Suite components using RMI connections. This adapter supports the propagation of both identities and transactions across JVMs and uses the T3 optimized path. Both synchronous and asynchronous invocation patterns are supported.
13	HTTP Binding Adapter	HTTP Binding Adapter enables the integration of SOA composite applications with HTTP binding; this enables the invocation of SOA-composite applications through HTTP POST and GET operations, and invokes HTTP end points through HTTP POST and GET operations.
14	Oracle Applications Adapter	Oracle Application Adapter enables integration with Oracle application components. This is a very important adapter that will aid in connecting with Oracle Applications.

S.No	Name of the Adapter	Description
15	Web Service Adapter	Web Service Adapter helps in integrating web services with Oracle SOA Suite. This is the replacement for partner links in 10g. All the web service references are creating using Web Service Adapter.
16	Third Party Adapter	Custom Adapters can be created using the Oracle adapter extensions framework and these adapters can be integrated with Oracle SOA Suite using the third-party adapter extensions.

## Applications adapters: E-Biz Suite, PeopleSoft, Siebel, and SAP

The Oracle Application adapter is available for SAP, Siebel, JD Edwards, and PeopleSoft. The following diagram depicts the architecture of the Application adapter's framework:



**Application Explorer:** this is a Java Swing-based design-time tool for configuring the packaged application adapters. Application Explorer provides the required features to enlist the application objects from the application and expose them service.

**BSE:** this is Business Service Engine, which is integrated as part of the Oracle Application adapter that will aid in communicating with the required packaged applications.

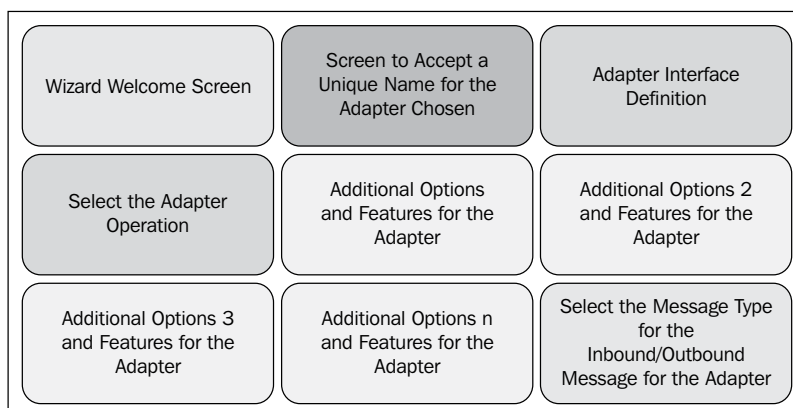
**SOAP Connections:** BSE uses SOAP connections as a protocol for accepting the client request and interacting with packaged applications.

## Adapter design-time configuration

The following is a list of adapter design-time configurations, which are explained in detail later on:

- File Adapter
- FTP Adapter
- MQ Series Adapter
- Socket Adapter
- Direct Binding Adapter
- HTTP Binding adapter
- DB Adapter
- AQ Adapter
- JMS Adapter
- B2B Adapter
- BAM Adapter

The following diagram shows the Adapter Configuration Wizard Flow:



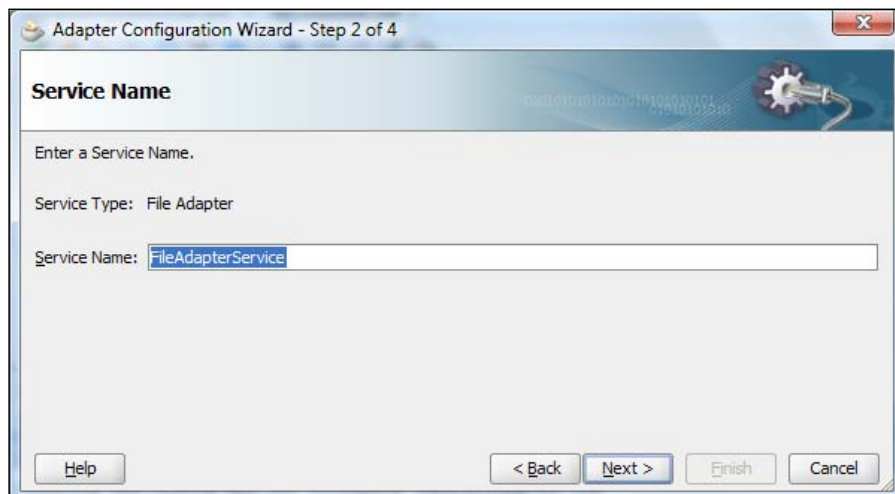
In the next section of this chapter, we will show the list of design-time configurations that are available for different adapters.

## File adapter configuration

1. The following screenshot shows the welcome screen for the **Adapter Configuration Wizard**:



2. Provide the **Service Name** for the adapter to be configured:



3. The **Adapter Interface** details are to be provided in this section of the wizard:

Adapter Configuration Wizard - Step 3 of 4

### Adapter Interface

The adapter interface is defined by a wsdl that is generated using the operation name and schema(s) specified later in this wizard. Optionally, the adapter interface may be defined by importing an existing WSDL.

Interface: ☒ Define from operation and schema (specified later)  
☐ Import an existing WSDL

WSDL URL:

Port Type:

Operation:

Help < Back Next > Finish Cancel

4. Choose the right operations to be used in the File Adapter configuration.  
**Write** for outbound operations and any other for inbound operations:

Adapter Configuration Wizard - Step 4 of 7

### Operation

The File Adapter supports four operations. There is a Read File operation that polls for incoming files in your local file system, a Write File operation that creates outgoing files, a Synchronous Read File operation that reads the current contents of a file, and a List Files operation that lists file names in specified locations. Specify the Operation type and Operation Name. Only one operation per Adapter Service may be defined using this wizard.

Operation Type: ☐ Read File  
☒ Write File  
☐ Synchronous Read File  
☐ List Files

Operation Name:

Help < Back Next > Finish Cancel

5. Specify the parameters to be used by the adapter for the operation selected:

Adapter Configuration Wizard - Step 5 of 7

### File Configuration

Specify the parameters for the Write File operation.

Directory specified as ☒ Physical Path ☐ Logical Name

Directory for Outgoing Files (physical path):  
C:\OracleSOACertification\Chapter\_04\_Working with Adapters Browse

File Naming Convention (po\_%SEQ%.txt): Sample\_File%SEQ%.txt

☐ Append to existing file

Write to output file when any of these conditions are met:

- ☒ Number of Messages Equals: 1
- ☐ Elapsed Time Exceeds: 1 minutes
- ☐ File Size Exceeds: 1000 kilobytes

Help < Back Next > Finish Cancel

6. Choose the message type to be used by the wizard:

Adapter Configuration Wizard - Step 6 of 7

### Messages

Define the message for the Write File operation. Specify the Schema File Location and select the Schema Element that defines the messages in the outgoing files. Use the Browse button to find an existing schema definition. If you check 'Schema is Opaque', then you do not need to specify a Schema.

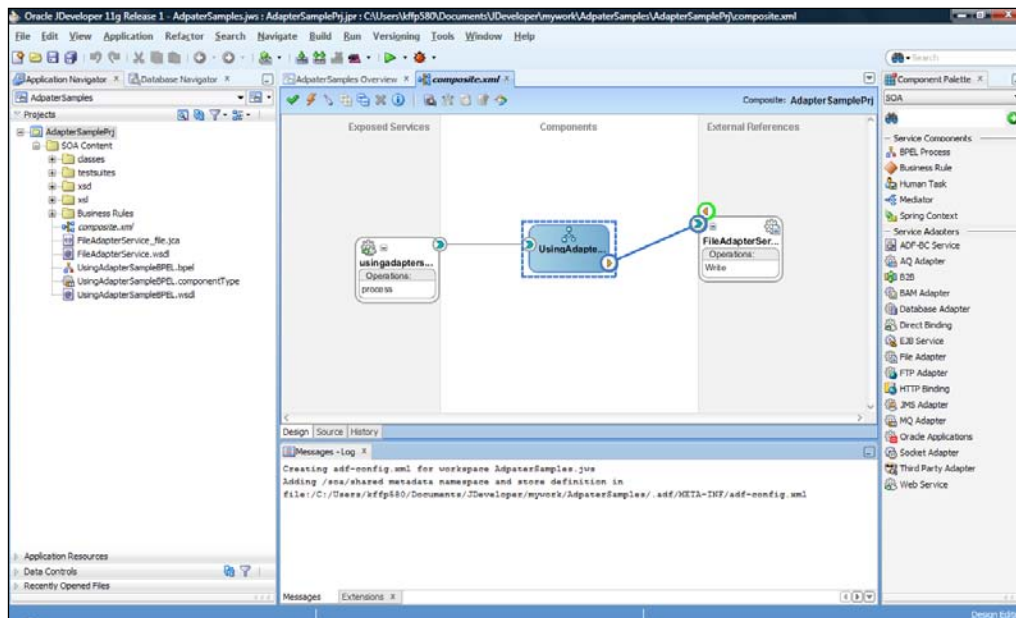
☐ Native format translation is not required (Schema is Opaque)

URL: xsd/UsingAdapterSampleBPEL.xsd Browse

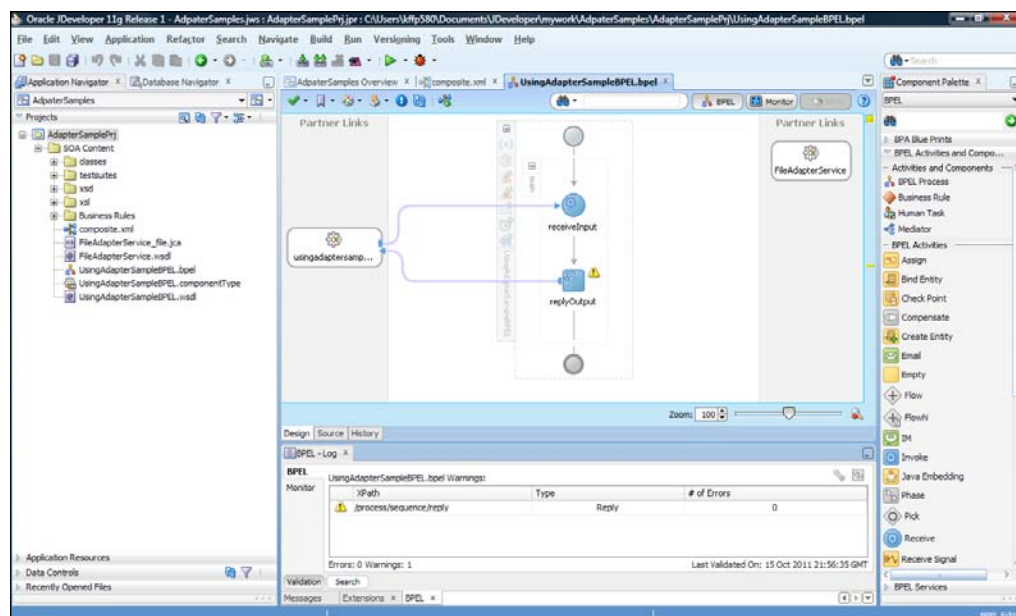
Schema Element: process specify URL for the schema (xsd) file

Help < Back Next > Finish Cancel

7. Wire the BPEL to connect to file adapter to create the required binding information in the `composite.xml` file:

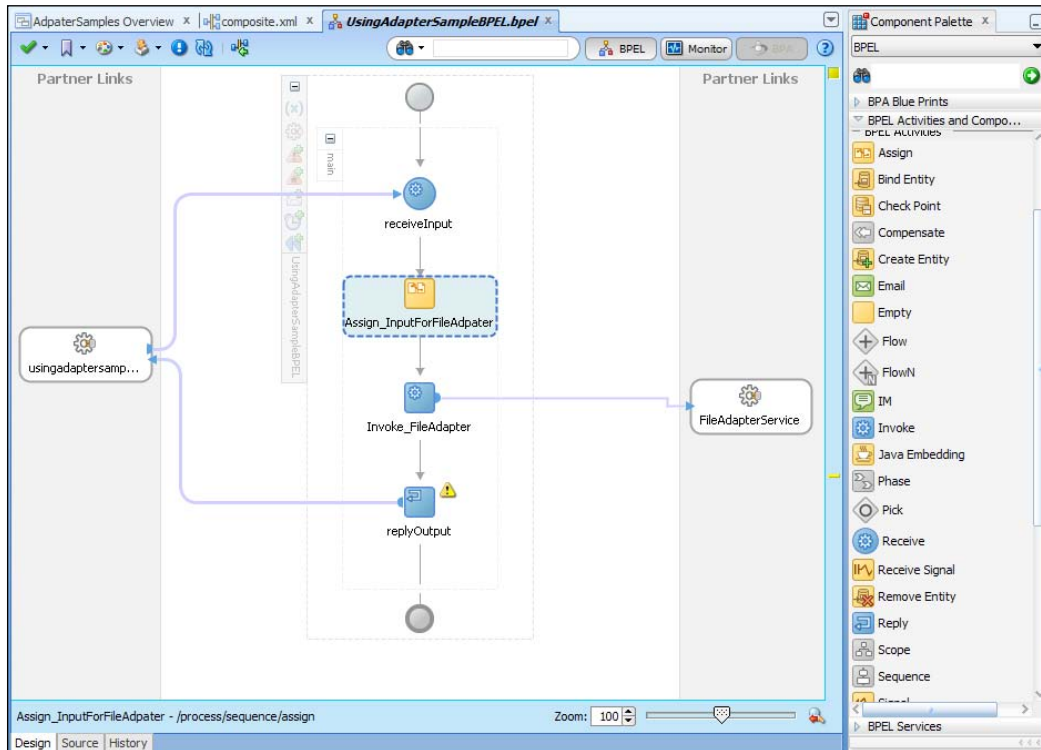


8. Invoke the file adapter from BPEL:





9. Assign the required input to be passed to the file adapter:



The wizard will create two key files that will be responsible for the adapter runtime to execute the configured instructions:

File Adapter configuration (FileAdapterService\_file.jca):

```
<adapter-config name="FileAdapterService" adapter="File Adapter" wsdlLocation="FileAdapterService.wsdl" xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
  <connection-factory location="eis/FileAdapter" adapterRef="" />
  <endpoint-interaction portType="Write_ptt" operation="Write">
    <interaction-spec className="oracle.tip.adapter.file.outbound.FileInteractionSpec">
      <property name="PhysicalDirectory" value="C:\OracleSOACertification\Chapter_04_Working with Adapters"/>
    </interaction-spec>
  </endpoint-interaction>
</adapter-config>
```

---

```

<property name="FileNamingConvention" value="Sample_File%SEQ%.txt" />
<property name="Append" value="false" />
<property name="NumberMessages" value="1" />
</interaction-spec>
</endpoint-interaction>
</adapter-config>

```

File adapter configuration (FileAdapterService.wsdl):

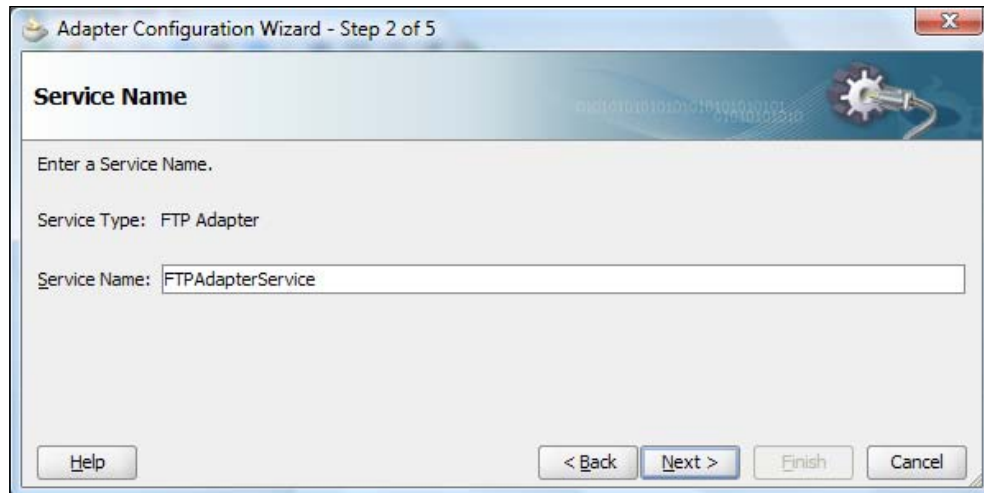
```

<?binding.jca FileAdapterService_file.jca?>
<wsdl:definitions name="FileAdapterService" targetNamespace="http://
xmlns.oracle.com/pcbpel/adapter/file/AdapterSamples/AdapterSamplePrj/
FileAdapterService" xmlns:tns="http://xmlns.oracle.com/pcbpel/
adapter/file/AdapterSamples/AdapterSamplePrj/FileAdapterService"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:impl="http://
xmlns.packt.com/AdapterSamples_jws/AdapterSamplePrj/
UsingAdapterSampleBPEL" xmlns:plt="http://schemas.xmlsoap.org/
ws/2003/05/partner-link/">
<plt:partnerLinkType name="Write_plt">
<plt:role name="Write_role">
<plt:portType name="tns:Write_ptt" />
</plt:role>
</plt:partnerLinkType>
<wsdl:types>
<schema xmlns="http://www.w3.org/2001/XMLSchema">
<import namespace="http://xmlns.packt.com/AdapterSamples_jws/
AdapterSamplePrj/UsingAdapterSampleBPEL" schemaLocation="xsd/
UsingAdapterSampleBPEL.xsd" />
</schema>
</wsdl:types>
<wsdl:message name="Write_msg">
<wsdl:part name="body" element="impl:process" />
</wsdl:message>
<wsdl:portType name="Write_ptt">
<wsdl:operation name="Write">
<wsdl:input message="tns:Write_msg" />
</wsdl:operation>
</wsdl:portType>
</wsdl:definitions>

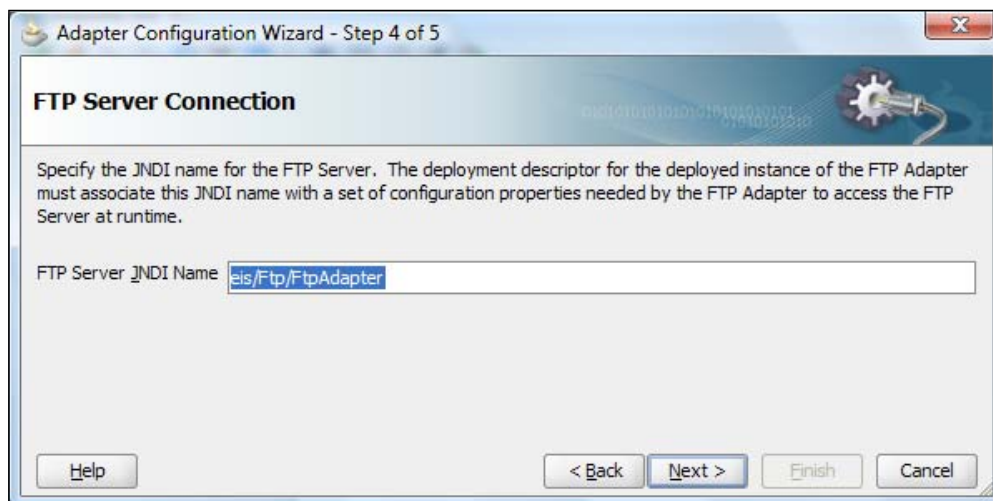
```

## FTP adapter configuration

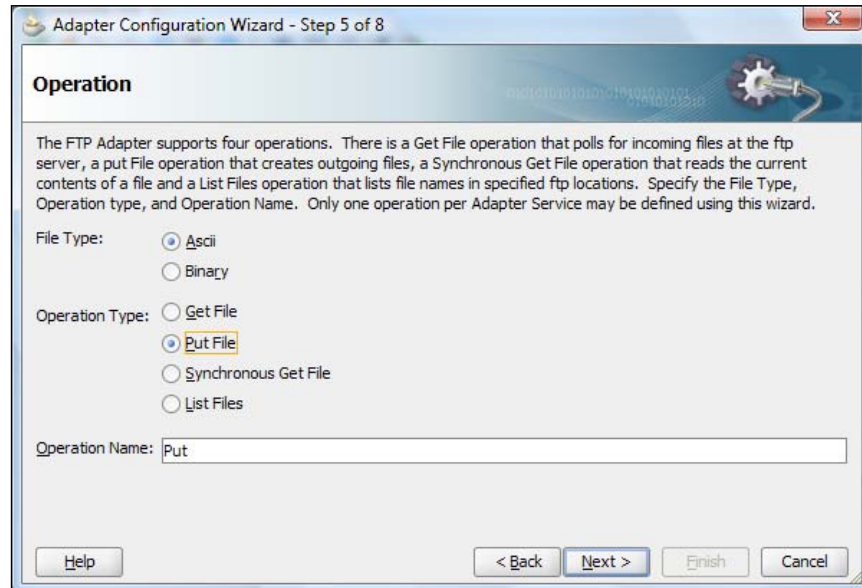
1. Get the FTP Adapter **Service Name**, as shown in the following screenshot:



2. Provide the **FTP Server JNDI Name**:



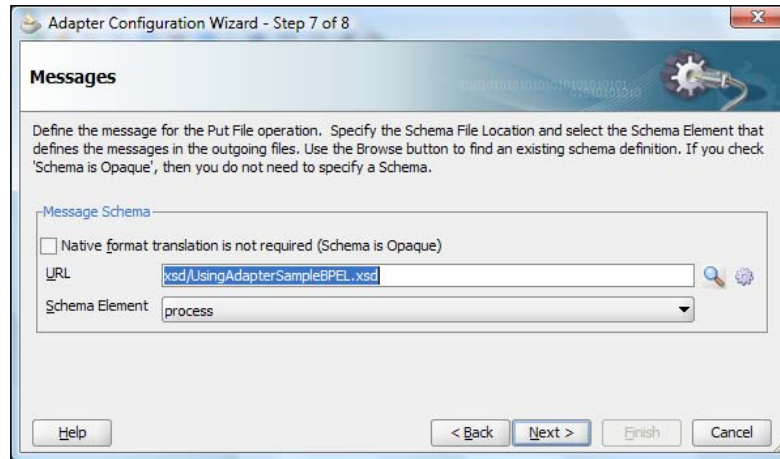
3. Choose the **Operation** to be used in the FTP Adapter Configuration Wizard:



4. Provide the required configuration to be used in configuring the FTP Adapter:



5. Enter the message type to be used for the inbound message of the adapter:



FTP adapter configuration (FTPAdapterService\_ftp.jca):

```
<adapter-config name="FTPAdapterService" adapter="Ftp Adapter"
wsdlLocation="FTPAdapterService.wsdl" xmlns="http://platform.
integration.oracle/blocks/adapter/fw/metadata">
<connection-factory location="eis/Ftp/FtpAdapter" adapterRef="" />
<endpoint-interaction portType="Put_ptt" operation="Put">
<interaction-spec className="oracle.tip.adapter.ftp.outbound.
FTPInteractionSpec">
<property name="FileType" value="ascii"/>
<property name="PhysicalDirectory" value="C:\OracleSOACertification\
Chapter_04_Working with Adapters"/>
<property name="FileNamingConvention" value="FTP_%SEQ%.txt"/>
<property name="Append" value="false"/>
<property name="NumberMessages" value="1"/>
</interaction-spec>
</endpoint-interaction>
</adapter-config>
```

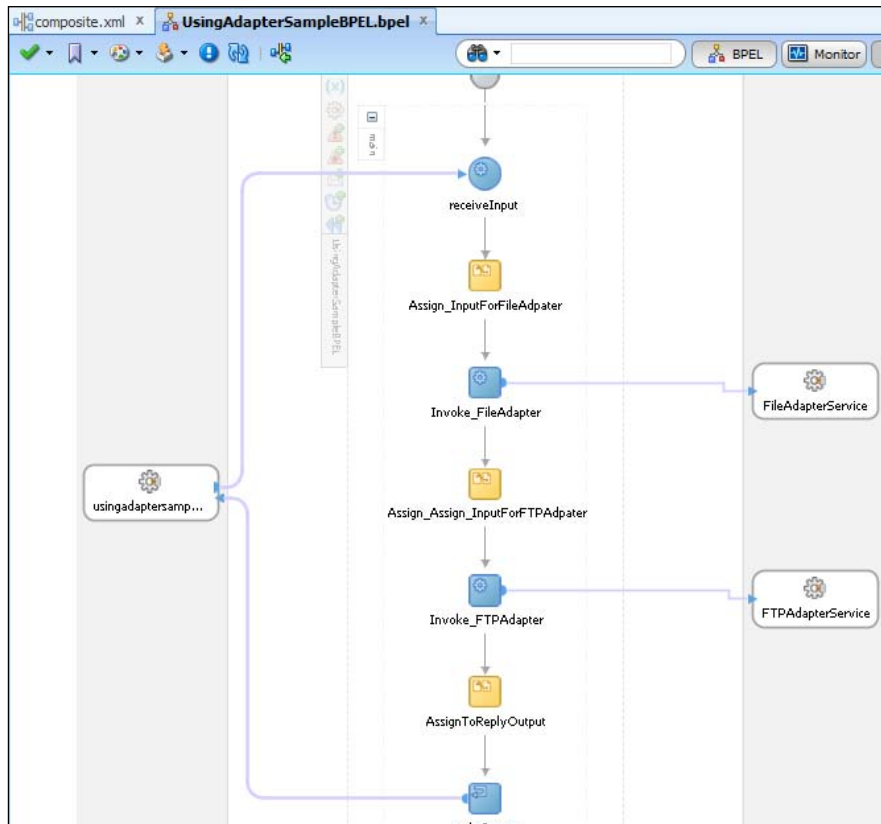
FTP adapter configuration (FTPAdapterService.wsdl):

```
<?binding.jca FTPAdapterService_ftp.jca?>
<wsdl:definitions name="FTPAdapterService" targetNamespace="http://
xmlns.oracle.com/pcbpel/adapter/ftp/AdapterSamples/AdapterSamplePrj/
FTPAdapterService" xmlns:tns="http://xmlns.oracle.com/pcbpel/
adapter/ftp/AdapterSamples/AdapterSamplePrj/FTPAdapterService"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:impl="http://
xmlns.pactk.com/AdapterSamples_jws/AdapterSamplePrj/
UsingAdapterSampleBPEL" xmlns:plt="http://schemas.xmlsoap.org/
ws/2003/05/partner-link/">
<plt:partnerLinkType name="Put_plt">
```

```

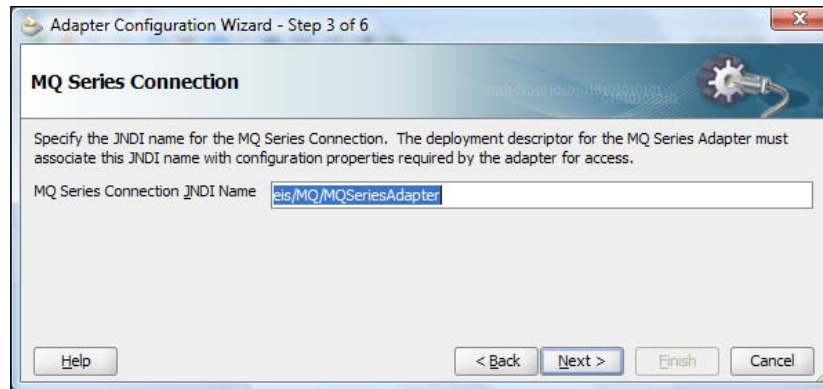
<plt:role name="Put_role">
<plt:portType name="tns:Put_ptt"/>
</plt:role>
</plt:partnerLinkType>
<wsdl:types>
<schema xmlns="http://www.w3.org/2001/XMLSchema">
<import namespace="http://xmlns.packt.com/AdapterSamples_jws/
AdapterSamplePrj/UsingAdapterSampleBPEL" schemaLocation="xsd/
UsingAdapterSampleBPEL.xsd"/>
</schema>
</wsdl:types>
<wsdl:message name="Put_msg">
<wsdl:part name="body" element="impl:process"/>
</wsdl:message>
<wsdl:portType name="Put_ptt">
<wsdl:operation name="Put">
<wsdl:input message="tns:Put_msg"/>
</wsdl:operation>
</wsdl:portType>
</wsdl:definitions>

```

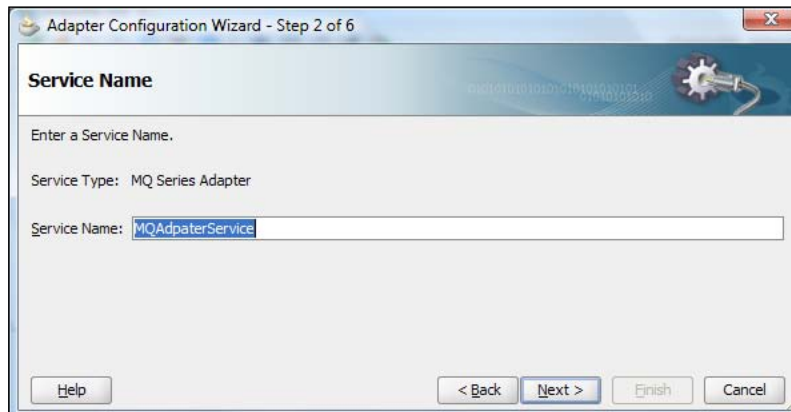


## MQ series adapter configuration details

1. Provide the **Service Name** of the adapter, as shown in the following screenshot:



2. Provide **JNDI Connection** details for the MQ Series server connection:



3. Define the **Adapter Interface** details:

Adapter Configuration Wizard - Step 4 of 6

### Adapter Interface

The adapter interface is defined by a wsdl that is generated using the operation name and schema(s) specified later in this wizard. Optionally, the adapter interface may be defined by importing an existing WSDL.

Interface: ☒ Define from operation and schema (specified later)  
☐ Import an existing WSDL

WSDL URL:

Port Type:

Operation:

Callback Port Type:

Callback Operation:

Help < Back Next > Finish Cancel

4. Choose the **Operation Type** to be used in configuring the MQ Adapter:

Adapter Configuration Wizard - Step 5 of 7

### Operation Type

Select an operation and specify an operation name, only one operation per Adapter Service may be defined using this wizard.

☒ Put Message into MQ  
☐ Get Message from MQ  
☐ Send Message to MQ and Get Reply/Reports  
☐ Get Message from MQ and Send Reply/Reports

Operation Name:

Help < Back Next > Finish Cancel



5. Provide the MQ queue details, as shown in the following screenshot:

Adapter Configuration Wizard - Step 6 of 7

### Put Message into MQ

Enter information for putting a normal message into MQ Series.

Queue Name	Queue Manager (optional)
SampleMQ	

☐ Partial Delivery

Message Format:

Priority:

Persistence:

Delivery Failure:

☐ Allow message to be segmented when necessary

Expiry:

☒ Never

☐ Expires in

6. Provide details for certain **Advanced Options**:

Adapter Configuration Wizard - Step 7 of 9

### Advanced Options

Specify any additional advanced options. To use the recommended, default values, click Next.

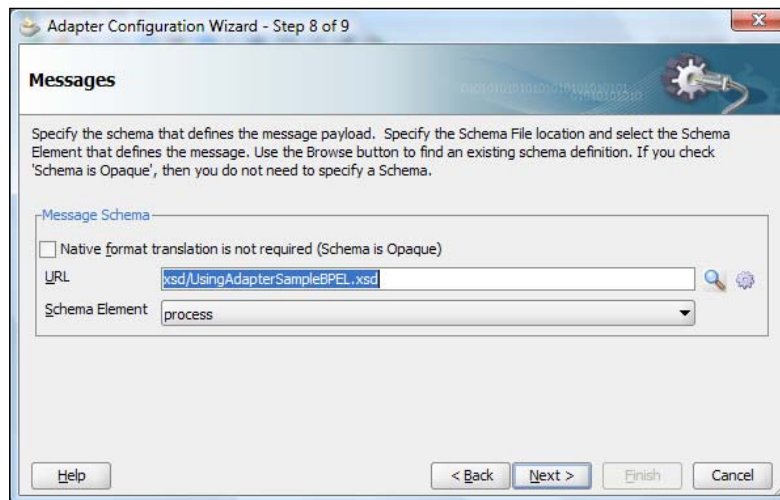
Auto-Retries:

Attempts:

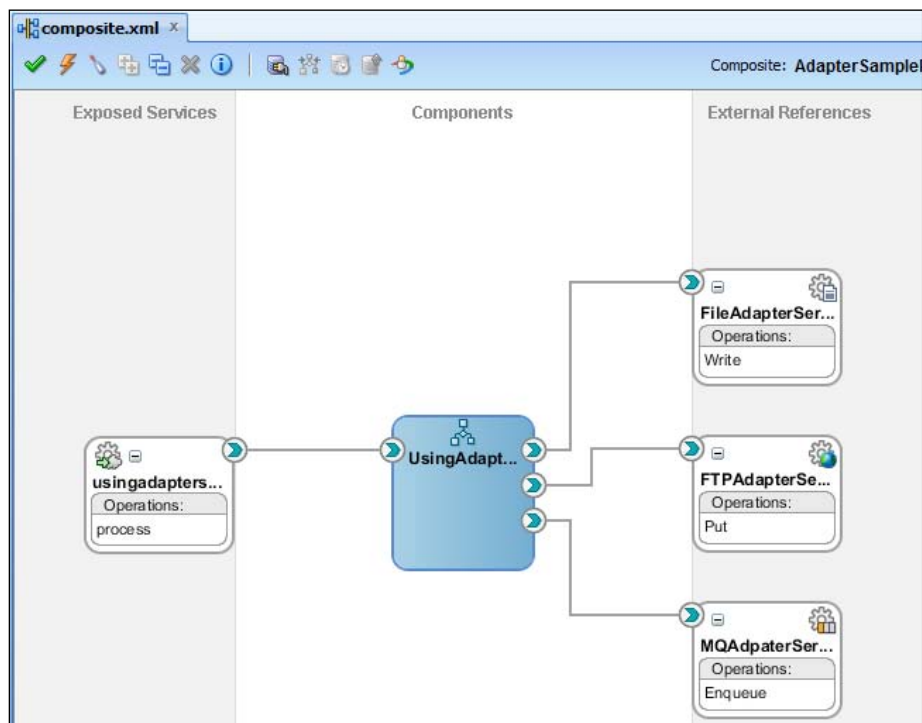
Interval (s):

Backoff Factor: x

- Specify the message type to be used by the adapter, as shown in the following screenshot:

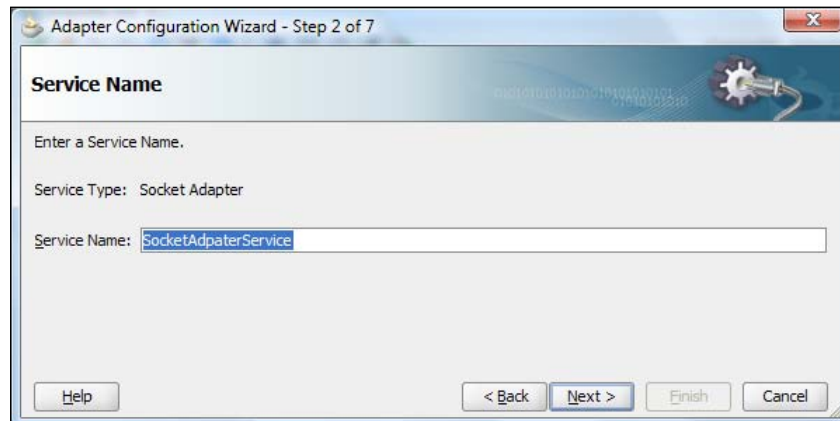


- Wire the MQ adapter to BPEL:



## Socket adapter service configuration

1. Provide the **Service Name** for the socket adapter, as shown in the following screenshot:



Adapter Configuration Wizard - Step 2 of 7

**Service Name**

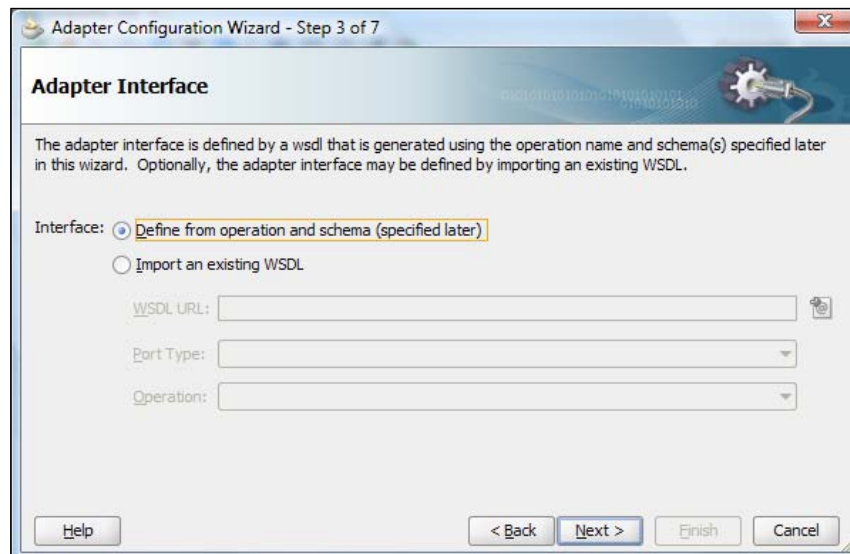
Enter a Service Name.

Service Type: Socket Adapter

Service Name:

Help < Back Next > Finish Cancel

2. Provide the **Adapter Interface** details, as shown in the following screenshot:



Adapter Configuration Wizard - Step 3 of 7

**Adapter Interface**

The adapter interface is defined by a wsdl that is generated using the operation name and schema(s) specified later in this wizard. Optionally, the adapter interface may be defined by importing an existing WSDL.

Interface: ☒ Define from operation and schema (specified later) ☐ Import an existing WSDL

WSDL URL:

Port Type:

Operation:

Help < Back Next > Finish Cancel

3. Choose the right **Operation Type** to be used in the Adapter Configuration Wizard:

Adapter Configuration Wizard - Step 4 of 7

### Operation Type

Select an operation and specify the operation name. Only one operation per Adapter Service may be defined using this wizard.

Operation Type: ☒ Inbound Synchronous Request/Reply  
☐ Inbound Receive  
☐ Outbound Synchronous Request/Reply  
☐ Outbound Invoke

Operation Name:

Help < Back Next > Finish Cancel

4. Provide the **Socket Connection** details:

Adapter Configuration Wizard - Step 5 of 7

### Socket Connection

Specify the JNDI name for the Socket Connection. The deployment descriptor for the Socket Adapter must associate this JNDI name with configuration properties required by the adapter for access.

Socket Connection JNDI Name:

☒ Specify Host and Port:

HostName:   
PortNumber:

Help < Back Next > Finish Cancel

5. Provide the **Message** details, as shown in the following screenshot:

**Adapter Configuration Wizard - Step 6 of 7**

**Messages**

Specify the schema that defines the message payload. Specify the Schema File location and select the Schema Element that defines the message. Use the Browse button to find an existing schema definition. If you check 'Schema is Opaque', then you do not need to specify a Schema.

**Request Message Schema**

☐ Native format translation is not required (Schema is Opaque)

URL: xsd/UsingAdapterSampleBPEL.xsd

Schema Element: process

**Reply Message Schema**

☐ Native format translation is not required (Schema is Opaque)

URL: xsd/UsingAdapterSampleBPEL.xsd

Schema Element: processResponse

Help < Back Next > Finish Cancel

**Adapter Configuration Wizard - Step 7 of 7**

**Protocol**

Specify the way you want to define the handshake(socket communication) steps.

☐ Use XSLT to define the handshake

Xslt:

ReplyXslt:

☐ Use Custom Java Code to define the handshake

Java Class:

☒ No Handshake

**Encoding/ByteOrder**

Specify the Encoding and Byte Order values. Only applicable if you are using translation.

☒ Specify Encoding/ByteOrder:

☐ Use Encoding Value from the Schema

☒ Encoding: ASCII

☐ Use Byte Order Value from the schema

☒ ByteOrder: BIG\_ENDIAN

Help < Back Next > Finish Cancel

## Direct Binding

**Direct Binding** is an option to directly integrate with other Oracle SOA Composite Services through RMI connections instead of web service calls. This optimizes the performance of the call between service components with the same SCA-INFRA.

This feature is also extended to Oracle Service Bus, hosted on the same WebLogic container.

The screenshot shows the 'Create Direct Binding' dialog box. The title bar says 'Create Direct Binding'. Inside, the 'Direct Binding' section has the instruction 'Create a Direct Binding.' and a gear icon. The form fields are as follows:

- Name:** DReference1
- Type:** Reference (dropdown)
- Reference Target:** Oracle SOA Composite (dropdown menu is open showing 'Oracle SOA Composite' and 'Oracle Service Bus')
- WSDL URL:** (text field)
- Port Type:** (dropdown)
- Callback Port Type:** (dropdown)
- ☐ Use SSL For Callback

Below these is the 'Reference Binding Details' section with two text fields:

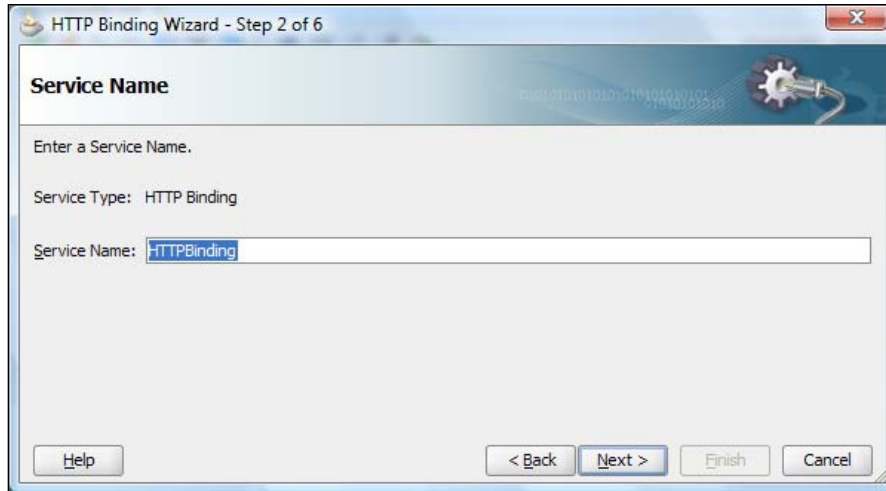
- Address:**
- Provider URL:**

At the bottom, there is a checkbox 'copy wsdl and its dependent artifacts into the project.' with a note: 'Note: Keeping a copy of a WSDL may result in synchronization issues if the remote WSDL is updated. It is recommended not make local copies - this should be reserved for situations such as offline designing.'

Buttons at the bottom: Help, OK, Cancel.

## HTTP binding wizard

1. Provide the **Service Name** details, as shown in the following screenshot:



HTTP Binding Wizard - Step 2 of 6

**Service Name**

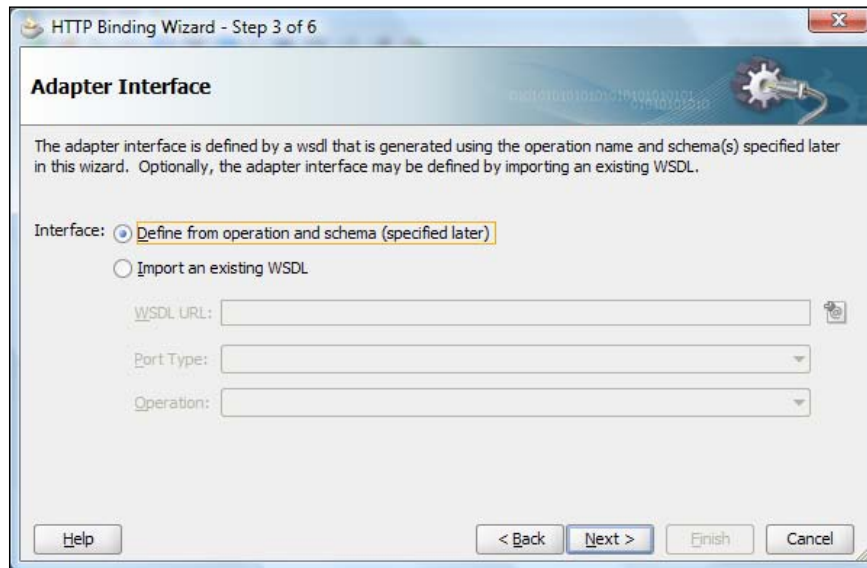
Enter a Service Name.

Service Type: HTTP Binding

Service Name:

Help < Back Next > Finish Cancel

2. Provide the **Adapter Interface** details, as shown in the following screenshot:



HTTP Binding Wizard - Step 3 of 6

**Adapter Interface**

The adapter interface is defined by a wsdl that is generated using the operation name and schema(s) specified later in this wizard. Optionally, the adapter interface may be defined by importing an existing WSDL.

Interface: ☒ Define from operation and schema (specified later) ☐ Import an existing WSDL

WSDL URL:

Port Type:

Operation:

Help < Back Next > Finish Cancel



3. Provide the **HTTP Binding Configuration**:

**HTTP Binding Configuration**

The HTTP Binding Adapter supports two operation types. There is a one-way operation type that sends or receives messages from an HTTP(s) endpoint, and a request-response operation type that sends and receives input and output messages to and from an HTTP(s) endpoint.

Type:

Operation Type: ☐ One-way ☒ Request-Response

Operation Name:

Verb:

Payload Type:

Endpoint:



Help < Back Next > Finish Cancel

## 4. Provide the message type details for the adapter configuration:

**Messages**



Specify the schema that defines the payload of the Http message. Specify the Schema File location and select the Schema Element that defines the message. Use the Browse button to find an existing schema definition or use the Define Schema button to define a schema for name-value pairs.

**Request Message Schema**

URL:   

Schema Element:

**Reply Message Schema**

URL:   

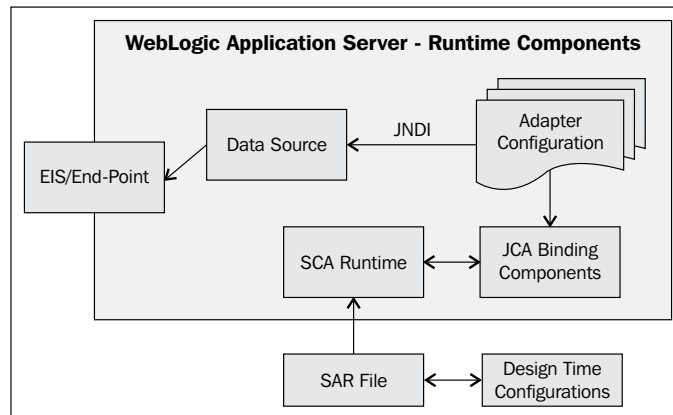
Schema Element:

Help < Back Next > Finish Cancel

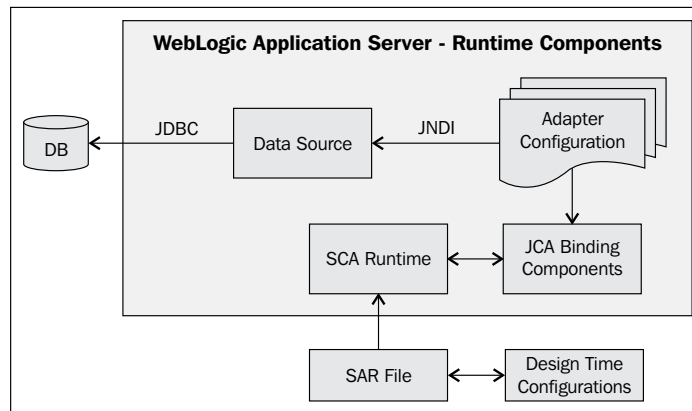


## Explain the runtime configuration

Runtime configuration for any adapter follows the following pattern, for most of the adapters. Runtime configuration is not required for the file adapters.



The following diagram shows the runtime configuration of the Oracle DB adapter using the adapter configurations:



## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Describe the adapter concepts and framework:** We discussed in detail about the **Java Connector Architecture (JCA)** and Oracle SOA Suite adapter framework
- **Describe Technology adapters:** We understood in detail about the various technology adapters such as File, Database, and JMS. We understood different patterns in using adapter as a service and as a reference
- **Describe Applications adapters:** We understood in detail about the Application adapters such as E-Biz, PeopleSoft, Siebel, and so on
- **Explain adapter design-time configuration:** We understood in detail about the design-time configuration of various technology and application adapters
- **Explain adapter runtime configuration:** We understood in detail about the runtime configuration functions and features of Oracle adapters

## Self-review questions

1. Adapter Configuration Wizard creates WSDL files for only the request-response service of an adapter.
  - a. True
  - b. False
2. Oracle FTP and File Adapter do not support XA transaction.
  - a. True
  - b. False
3. Identify which of the following adapters support XA transaction.
  - a. JMS Adapter
  - b. AQ Adapter
  - c. Database Adapter
  - d. All the above

4. Support for Metadata Service in Oracle adapters is not found in Oracle SOA Suite 11g.
  - a. True
  - b. False
5. Rejected Messages Handling for inbound adapters are managed through
  - a. Properties specified in `composite.xml`
  - b. Inbound Rejected Messages are handled through fault-policies
  - c. Inbound Rejected Messages are automatically handled by SOA-INFRA and no configuration is required
  - d. None of the above
6. Oracle Adapters support large payload support in \_\_\_\_\_.
  - a. Oracle File adapter
  - b. Socket adapter
  - c. HTTP Binding
  - d. Oracle Apps adapter
7. Batching and de-batching features are available in the \_\_\_\_\_ adapter.
  - a. File adapter
  - b. FTP adapter
  - c. Database adapter
  - d. All the above
8. Which statement is true regarding XA transaction with an Oracle SCA Component?
  - a. SCA Component should initiate the XA transaction to make the adapter participate in it
  - b. SCA Component invoking the adapter should participate in an XA transaction
  - c. Inbound adapters cannot participate in the XA transaction
  - d. By default, in Oracle SOA Suite, all adapters are XA transaction-enabled

9. The Fault Policy mechanism doesn't work for which type of transaction?
  - a. Outbound adapters in XA mode
  - b. Outbound adapter in Mediator Sequential Routing
  - c. Both a and b
  - d. None of the above
10. Oracle File adapter has the feature to delete the files after processing the files from the source directories.
  - a. True
  - b. False



# 5

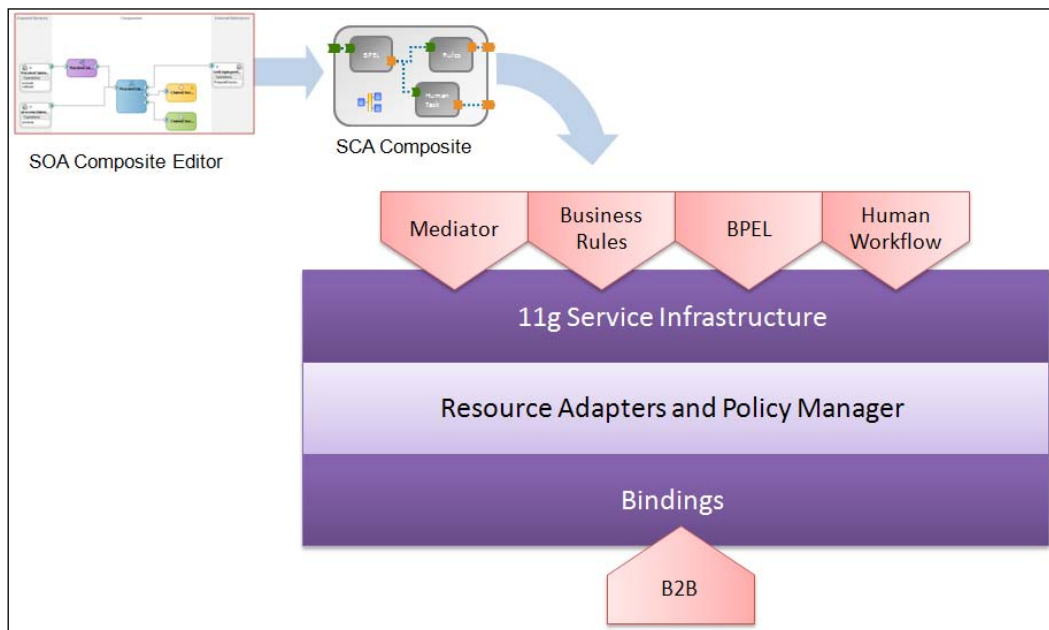
## Orchestrating Services with BPEL

**BPEL** expands to **Business Process Execution Language**. BPEL is an essential tool in the SOA world to achieve the **Business Process Orchestration**. It is a very thin and light-weight XML-based language that can be used to accomplish the Process Orchestration. BPEL uses Web Services exposed from different business applications and information systems (through technology adapters) to achieve the Process Orchestration.

OASIS drafted BPEL standards by working closely with various industry-leading companies such as IBM, Oracle, Amber Point, SAP, SOA Systems, iWay Software, and so on. It is predicted that BPEL will be the next widely used language in most Enterprises after Java. In Oracle SOA Suite 11g, Oracle BPEL Process Manager is integrated to Oracle SCA Runtime. This helps the Oracle SCA runtime to integrate the various Oracle SOA Components together to accomplish some very interesting tasks. Oracle BPEL Process Manager, introduced in 10g, is now integrated into Oracle SCA Runtime Components as a process engine along with Mediator, Business Rule, and Human Workflow Engines. Oracle SOA Suite 11g (11.1.1.5) supports both BPEL 1.x and 2.0 features.

## What BPEL helps to accomplish

Orchestration might sound like jargon but is it so? Yes, languages used to communicate with computers are always complex. Until recently, the software industry has not created a programming language that does all the required computing functionality. You can imagine how complex it will be for a language to interface with multiple platforms and applications that are evolved using different languages. After a decade long research and introduction of various industry standard protocols, BPEL evolved to achieve this. BPEL is a coordinator that can talk to multiple technologies and accomplish the required business process functionality that cuts across different Enterprise Business Applications. BPEL supports both synchronous and asynchronous interactions.



## Is BPEL a game changing technology?

**BPasS (Business Process as a Service)** is another buzzword in the industry today, and it is all set to take its position in cloud based computing. BPEL will be the tool that will be used to achieve this. BPEL is the first kind of technology/programming language that uses the database to hydrate the state of the process. This feature is a clear differentiator from other persistent technologies that are currently known. This feature helps in executing long-running business processes in the organization and maintains the process state without hitting the in-memory constraints.

This chapter will guide the readers and help them to understand various features of Oracle BPEL Process Engine and BPEL Component using Oracle SOA Suite 11g and prepare them to meet the exam objectives listed as follows:

- Describe synchronous and asynchronous BPEL processes
- Explain BPEL components, activities, and partner links
- Explain parallel flow and conditional branching
- Explain BPEL integration with Java

## Synchronous and asynchronous BPEL processes

As we discussed, a BPEL process is used to communicate with different enterprise applications and it supports different types of communication. The following is a list of communication techniques that are provided by BPEL:

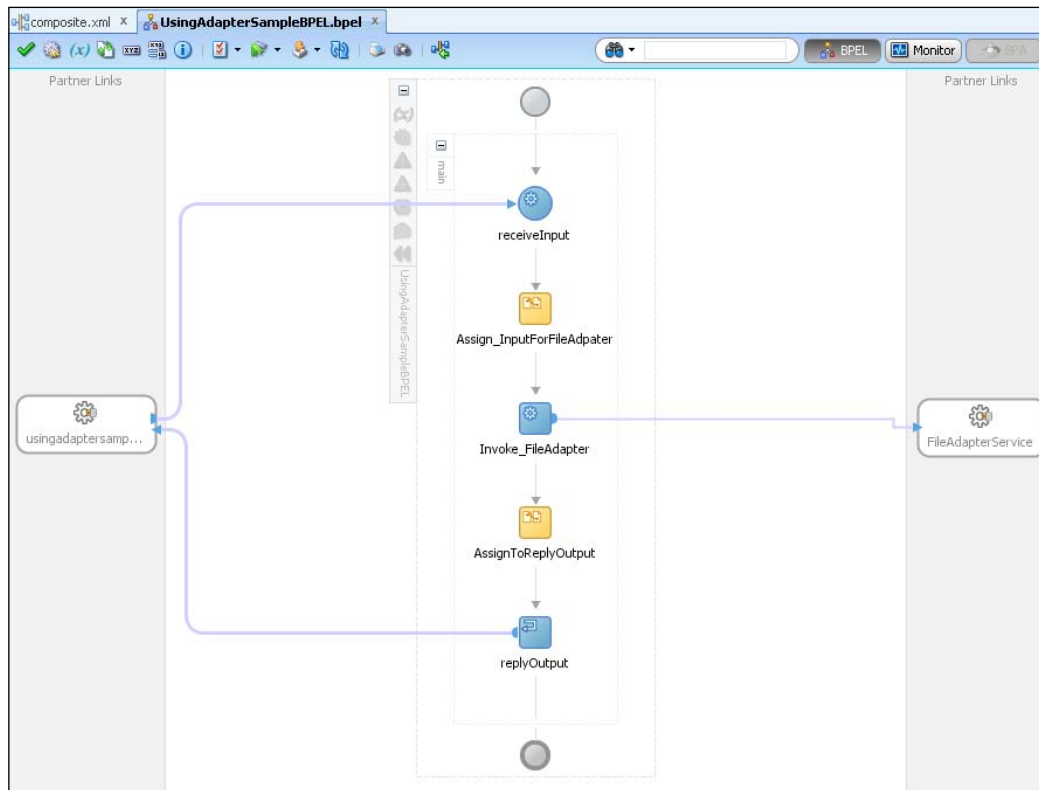
- Synchronous
- Asynchronous – one way
- Asynchronous – request – response (undefined wait period)

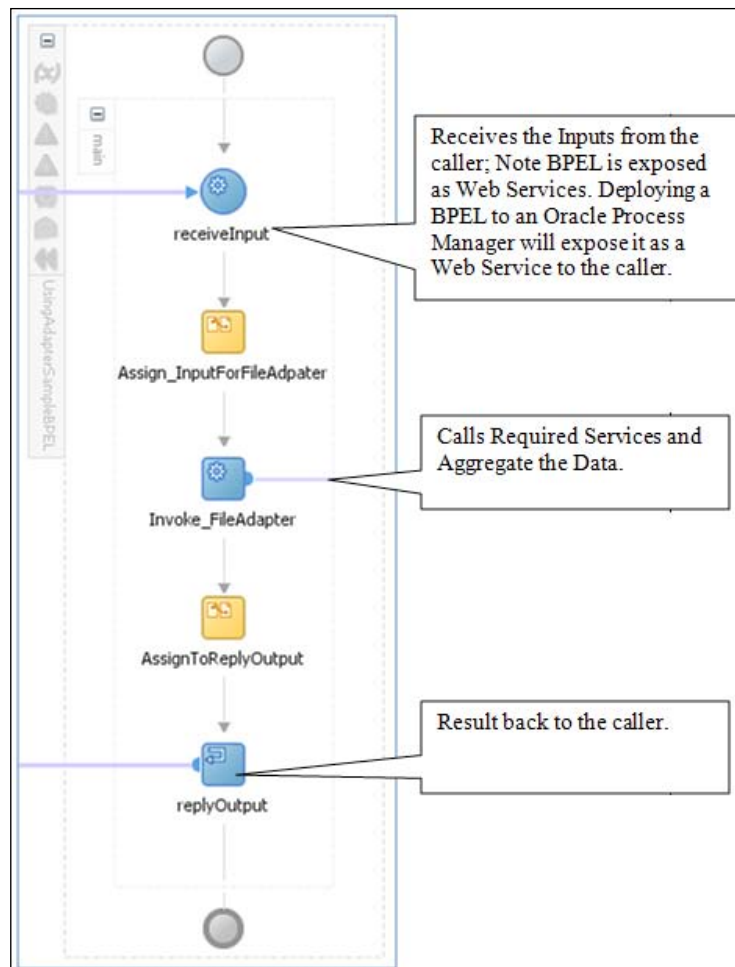


## Synchronous

**Synchronous BPEL** process template helps to build services that respond for the given request. It helps to achieve communication between different systems to provide an aggregated response with a predefined threshold to respond to the caller.

The following images show the synchronous BPEL processes:





Key information to be noted to identify the BPEL as synchronous is as follows:

- It uses only one port to communicate
- It uses input, output, and fault messages to communicate with the caller

The following WSDL is generated by the Oracle JDeveloper for a synchronous BPEL process:

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="UsingAdapterSampleBPEL"
targetNamespace="http://xmlns.packt.com/AdpaterSamples_jws/
AdapterSamplePrj/UsingAdapterSampleBPEL"
```

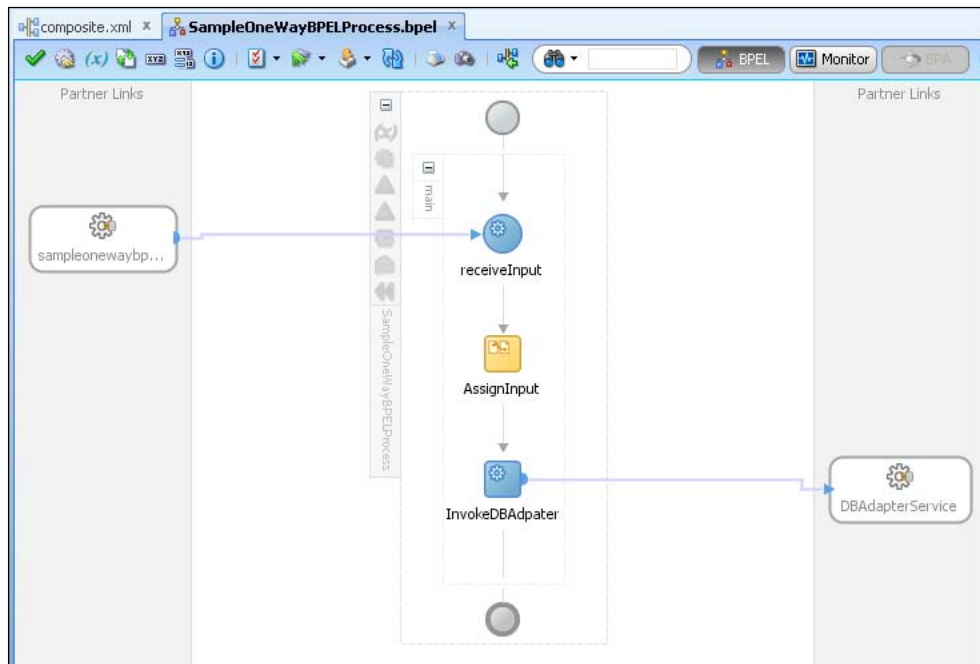
```
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:client="http://xmlns.packt.com/AdpaterSamples_jws/
AdapterSamplePrj/UsingAdapterSampleBPEL"
      xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/
partner-link/">
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import namespace="http://xmlns.packt.com/
AdpaterSamples_jws/AdapterSamplePrj/
UsingAdapterSampleBPEL" schemaLocation="xsd/UsingAdapterSampleBPEL.
xsd"/>
    </schema>
  </wsdl:types>

  <wsdl:message name="UsingAdapterSampleBPELRequestMessage">
    <wsdl:part name="payload" element="client:process"/>
  </wsdl:message>
  <wsdl:message name="UsingAdapterSampleBPELResponseMessage">
    <wsdl:part name="payload" element="client:processResponse"/>
  </wsdl:message>

  <!--portType implemented by the UsingAdapterSampleBPEL BPEL
process-->
  <wsdl:portType name="UsingAdapterSampleBPEL">
    <wsdl:operation name="process">
      <wsdl:input message="client:UsingAdapterSampleBPELRe
questMessage" />
      <wsdl:output message="client:UsingAdapterSampleBPELRe
sponseMessage"/>
    </wsdl:operation>
  </wsdl:portType>
  <plnk:partnerLinkType name="UsingAdapterSampleBPEL">
    <plnk:role name="UsingAdapterSampleBPELProvider">
      <plnk:portType name="client:UsingAdapterSampleBPEL"/>
    </plnk:role>
  </plnk:partnerLinkType>
</wsdl:definitions>
```

## Asynchronous – one-way

Asynchronous – one way BPEL process is used to call a target service or perform outbound operations such as writing to a file in a directory, FTP site, enqueue a message into a queue, or call a service. This process can be used to implement functionalities that do not require a response from the caller. You will also notice that reply activity is not available in asynchronous. One-way BPEL process template is shown in the following screenshot:



Key information to be noted in order to identify the BPEL as a one-way asynchronous BPEL process is as follows:

- It uses only one port to communicate
- It uses only input messages, does not require output messages as response, and it need not be sent back to the caller

The following WSDL is generated by the Oracle JDeveloper for a one-way asynchronous BPEL process:

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="SampleOneWayBPELProcess"
targetNamespace="http://xmlns.oracle.com/AdpaterSamples/
AdapterSamplePrj/SampleOneWayBPELProcess"
```

```
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:client="http://xmlns.oracle.com/AdpaterSamples/AdapterSamplePrj/
SampleOneWayBPELProcess"
xmlns:plnk="http://docs.oasis-open.org/wsbpel/2.0/plnktype">
  <wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
      <import namespace="http://xmlns.oracle.com/
AdpaterSamples/AdapterSamplePrj/SampleOneWayBPELProcess"
schemaLocation="xsd/SampleOneWayBPELProcess.xsd" />
    </schema>
  </wsdl:types>

  <wsdl:message name="SampleOneWayBPELProcessRequestMessage">
    <wsdl:part name="payload" element="client:process"/>
  </wsdl:message>

  <!--portType implemented by the SampleOneWayBPELProcess BPEL
process -->
  <wsdl:portType name="SampleOneWayBPELProcess">
    <wsdl:operation name="process">
      <wsdl:input message="client:SampleOneWayBPELProcessR
equestMessage" />
    </wsdl:operation>
  </wsdl:portType>

  <plnk:partnerLinkType name="SampleOneWayBPELProcess">
    <plnk:role name="SampleOneWayBPELProcessProvider" portType="
client:SampleOneWayBPELProcess"/>
  </plnk:partnerLinkType>
</wsdl:definitions>
```

## Asynchronous – bi-directional

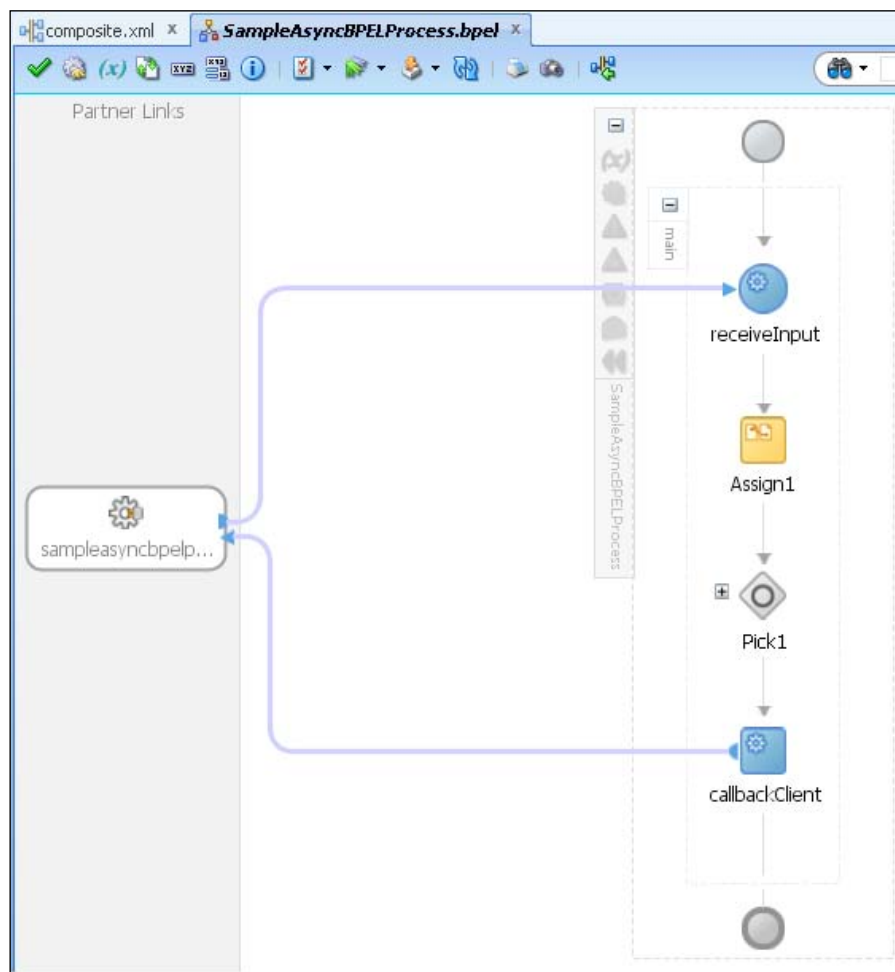
The following image shows a sample asynchronous bi-directional BPEL process. Asynchronous communication is a complicated communication technique compared to synchronous communication. An asynchronous process keeps track of the caller and sends the response to the caller on completion of the requested task. Calling the appropriate caller for the requested message is done through a technique called **correlation**. We will discuss, in detail, the correlation techniques in the next chapter.

Key points to be noted about an asynchronous process are as follows:

- Callback is a reply activity and it is not a special type of activity introduced for an asynchronous BPEL process. It is differentiated with a different name and representation to identify the asynchronous BPEL process.

- Long-running BPEL processes are implemented as asynchronous BPEL processes.
- The BPEL process manager dehydrates the BPEL process state when it encounters any of the following states in the BPEL process:
  - Wait activity
  - OnMessage
  - Pick
  - Checkpoint (forced dehydration step); it is renamed as Dehydrate in 11.1.1.5

These states are shown in the following image:



- Key information to be noted in order to identify the BPEL as asynchronous – bi-directional are as follows:
  - It uses two ports to communicate, one for receiving the message and another for responding to the message from the caller
  - It uses input and output messages to communicate with the caller

The following WSDL is generated by the Oracle JDeveloper for an asynchronous – bi-directional BPEL process:

```
<wsdl:types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema">
        <import namespace="http://xmlns.oracle.com/AdapterSamples/
AdapterSamplePrj/SampleAsyncBPELProcess" schemaLocation="xsd/
SampleAsyncBPELProcess.xsd" />
    </schema>
</wsdl:types>

<wsdl:message name="SampleAsyncBPELProcessRequestMessage">
    <wsdl:part name="payload" element="client:process"/>
</wsdl:message>

<wsdl:message name="SampleAsyncBPELProcessResponseMessage">
    <wsdl:part name="payload" element="client:processResponse"/>
</wsdl:message>

<wsdl:portType name="SampleAsyncBPELProcess">
    <wsdl:operation name="process">
        <wsdl:input message="client:SampleAsyncBPELProcessReq
uestMessage"/>
    </wsdl:operation>
</wsdl:portType>

<wsdl:portType name="SampleAsyncBPELProcessCallback">
    <wsdl:operation name="processResponse">
        <wsdl:input message="client:SampleAsyncBPELProcessRes
ponseMessage"/>
    </wsdl:operation>
</wsdl:portType>

<plnk:partnerLinkType name="SampleAsyncBPELProcess">
    <plnk:role name="SampleAsyncBPELProcessProvider">
        <plnk:portType name="client:SampleAsyncBPELProcess"/>
    </plnk:role>
    <plnk:role name="SampleAsyncBPELProcessRequester">
```

```

        <plnk:portType name="client:SampleAsyncBPELProcessCal
lback"/>
    </plnk:role>
</plnk:partnerLinkType>
</wsdl:definitions>

```

## BPEL components activities and Partner Links

Oracle Process Manager provides different types of components, which include the following:

- BPEL constructs that are defined as part of OASIS BPEL standards definition
- Oracle extensions to BPEL constructs
- BPEL services – we have discussed in detail the entire list of BPEL services in *Chapter 3, Working with Adapters*

The standard BPEL constructs are as follows:

S.No	Construct	Details
1	Invoke	Provides the feature to invoke the required partner link
2	Partner Link	Partner link is a Web Service or adapter that is called from the BPEL process
3	Receive	The Receive activity can be used to receive the message from the BPEL process. This can be used at the start or in the middle of the BPEL process.
4	Reply	The Reply activity is used to send the message back to the caller. This is used by the BPEL process in synchronous and asynchronous communication.
5	Assign	Assigns the variables to values or expressions.
6	Compensate	The Compensate activity is a business activity that compensates for the previous executed steps.
7	Empty	The Empty activity is used to represent no action required on a branch of Flow or While constructs.
8	Terminate	The Terminate activity ends the BPEL process execution.
9	Throw	The Throw activity throws the fault that is captured in the Catch block or Fault that is constructed in the CatchAll block.
10	Wait	The Wait activity provides features to pause the BPEL process execution for a specified period of time.



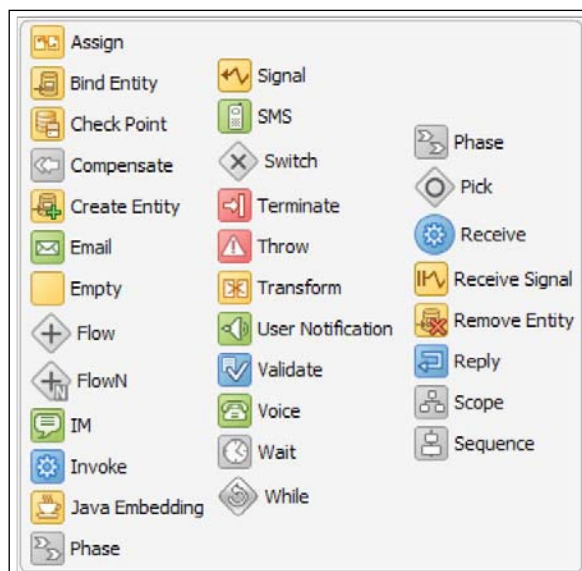
S.No	Construct	Details
11	Flow	The Flow activity is a unique activity for BPEL. It helps to run the tasks in parallel. There are two types of flow activities.
12	Pick	Pick is also a unique activity for BPEL. This activity is used to receive the message from various different partner links.  It is used in an asynchronous BPEL process. This can be used at the start or in the middle of the BPEL process as well.
13	Scope	Scope is an activity used to group the set of BPEL activities for logically grouping a set of actions to handle faults, define local variables for manipulation, and so on.
14	Sequence	Sequence is another grouping construct that is used to sequence the list of BPEL activities.
15	Switch	Switch is used very similarly to switch case constructs in C programming language.
16	While	While is a looping construct that is used to execute the list of actions until the condition specified in While constructs returns to false.
17	If (BPEL 2.0)	Switch is replaced with If in BPEL 2.0. This construct helps us perform the if-else operations.
18	Repeat Until (BPEL 2.0)	This is an additional Construct introduced in BPEL 2.0 to repeat certain actions until the condition is met.
19	ReThrow (BPEL 2.0)	This is an additional construct that allows the re-throw of the fault caught in the Fault handler block.
20	ForEach (BPEL 2.0)	This is an additional construct introduced in BPEL 2.0. This will replace the flown construct by allowing parallel execution.

The Oracle extensions to BPEL constructs are as follows:

S.No	Construct	Details
1	Email	Activity to send e-mail to any target recipient.
2	User Notification	Uses the UMS (Unified Messaging Framework) to send SMS, e-mail, IM, and Voice.
3	Validate	Validates a variable data against the schema definition.
4	IM	Sends an instance message to the required person.
5	Checkpoint	Checkpoint activity helps to dehydrate the BPEL process to the database.

S.No	Construct	Details
6	Phase	This is a special purpose activity which helps to construct an output message based on the business rules defined around the inputs message. This can be used as a replacement for data transformation where multiple condition checks are used.
7	Voice	This activity is used to send a telephone voice notification.
8	Signal	This activity is used in the master process to send a signal to the detailed process.
9	Receive Signal	This activity is used in the detailed process to receive a signal sent from the master process.
10	Create Entity	This activity is used to create an entity variable.
11	Bind Entity	This activity is used to bind an entity variable.
12	Transformation	This activity is used to call an XSLT function from BPEL.
13	Java Embedding	This activity enables the process designer to execute Java code to facilitate the reuse of existing Java code for processing logics. This is recommended to be used less frequently.

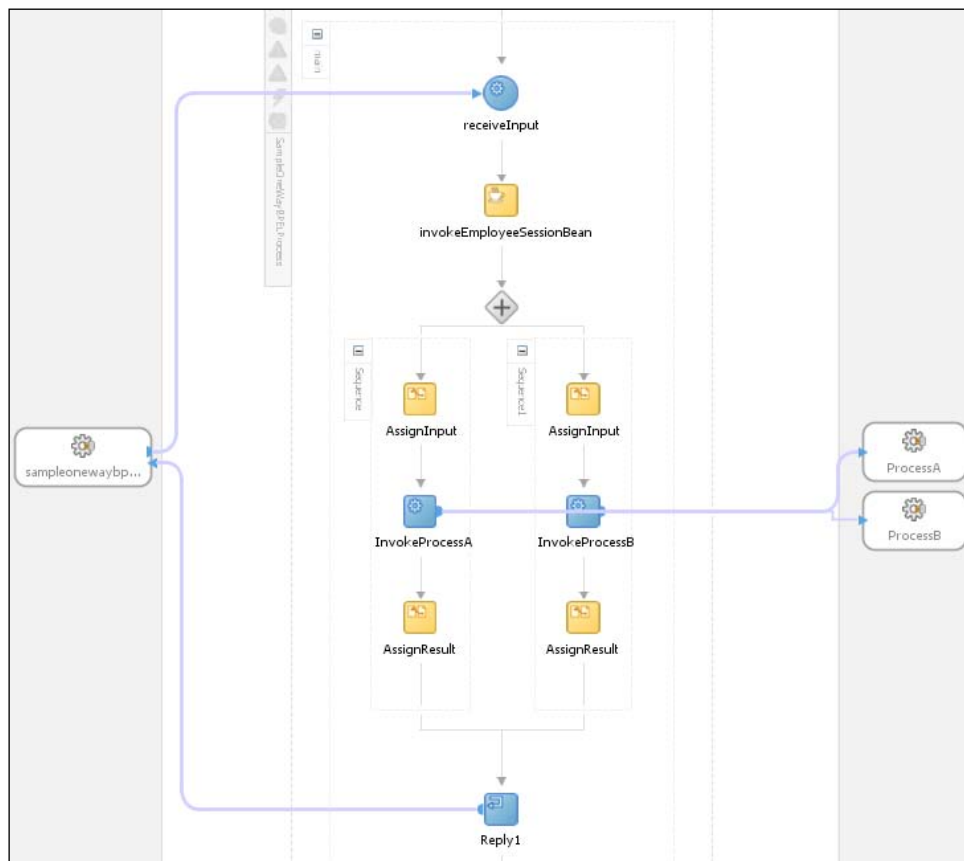
BPEL Constructs from JDeveloper 11.1.1.5 are shown in the following image:



## Parallel flow and conditional branching

Parallel flows can be achieved using the Flow and FlowN (ForEach in BPEL 2.0) activities. Flow activity will help to execute the BPEL activities in parallel. This will reduce the process time if independent activities are performed as a part of the same business process. Flow activity is typically used in scenarios where results from invoking multiple asynchronous processes are independent to each other. Flow activity will determine the number of flow arms during design time. Flow allows calling or executing different business logic simultaneously, but the order of execution is not guaranteed. FlowN allows true parallelism by calling upon the same set of business logic on different sets of data. The ForEach construct, introduced in BPEL 2.0, also allows the process to dynamically pick up partners and perform operations similar to FlowN.

The following image shows the Flow activity:



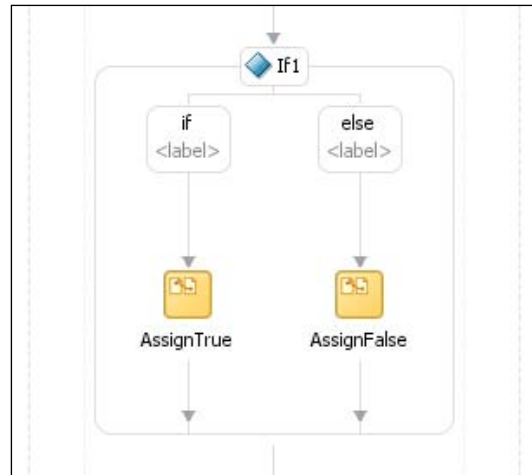
A code snippet showing the flow activity is as follows:

```
<flow name="Flow1">
  <sequence name="Sequence2">
    <assign name="AssignInput"></assign>
    <invoke name="InvokeProcessA" bpelx:invokeAsDetail="no"
partnerLink="ProcessA" portType="ns1:Request_Response_ptt"
operation="Request-Response" />
    <assign name="AssignResult"></assign>
  </sequence>
  <sequence name="Sequence1">
    <assign name="AssignInput"></assign>
    <invoke name="InvokeProcessB" bpelx:invokeAsDetail="no"
partnerLink="ProcessB" portType="ns2:execute_ptt"
operation="execute" />
    <assign name="AssignResult"></assign>
  </sequence>
</flow>
```

The code snippet: `forEach` activity is as follows:

```
<forEach parallel="yes" counterName="ForEach1Counter" name="ForEach1">
  <startCounterValue>count($inputVariable.payload/client:input)</
startCounterValue>
  <finalCounterValue/>
  <scope name="Scope1">
    <sequence name="Sequence2">
      <assign name="AssignInput"></assign>
      <invoke name="InvokeProcessA" bpelx:invokeAsDetail="no"
partnerLink="ProcessA" portType="ns1:Request_Response_ptt"
operation="Request-Response" />
      <assign name="AssignResult"></assign>
    </sequence>
  </scope>
</forEach>
```

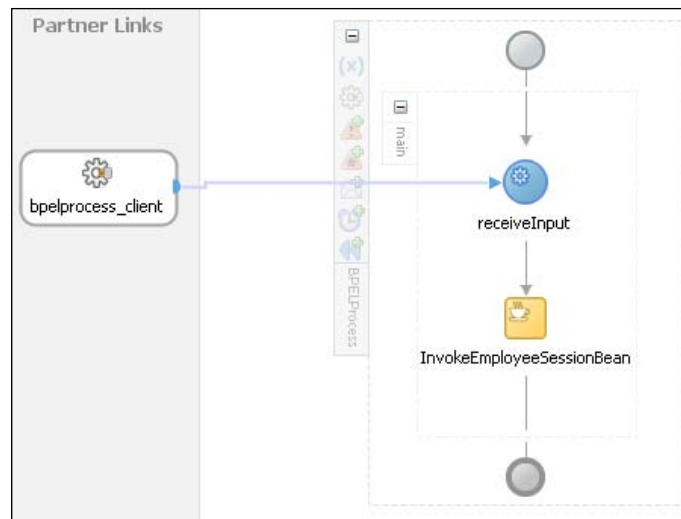
Conditional branching can be achieved using Switch and if-else constructs. Switch is a BPEL 1.x construct that is deprecated in BPEL 2.0.



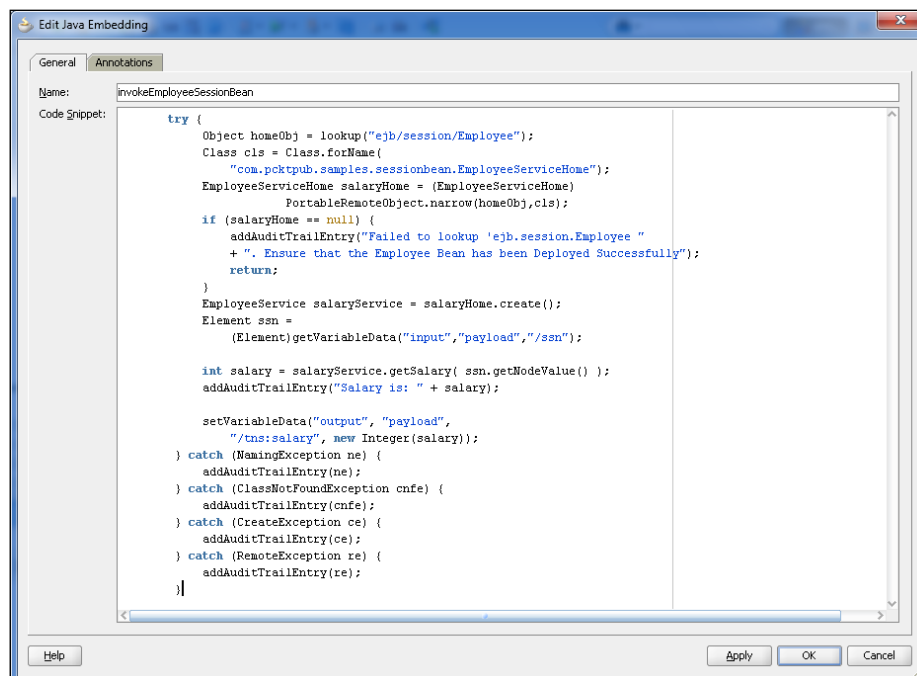
```
<if name="If1">
<documentation/>
<condition>$inputVariable.payload/client:input> 10</condition>
<assign name="AssignTrue"></assign>
<else>
<assign name="AssignFalse"></assign>
</else>
</if>
```

## **BPEL integration with Java**

Java code can be integrated with BPEL using the Oracle BPEL Extension Constructs. This construct helps to write a simple Java code to be used along with BPEL. This feature is very useful to reuse the rich set of Java libraries for various operations to be performed on the data.



The following screenshot shows the Java code that is used within the BPEL process to execute a session bean. It is highly recommended not to implement most of the business logic using Java Embedding and use BPEL to wrap it as a Web Service; there are complementary technologies available to do this. It is recommended that the user employs Java Embedding as a utility to extend the features from Java.



The following table illustrates the lists of `bpelx:exec` built-in methods that you can use to read and update scope variables, instance metadata, and audit trails:

S.No	Features	Function
1	JNDI access	<code>Object lookup(String name)</code>
2	Oracle BPEL process manager locator	<code>Locator getLocator()</code>
3	Unique ID associated with each instance	<code>long getInstanceId()</code>
4	Set the title of this BPEL instance	<code>String setTitle(String title)</code> <code>String getTitle()</code>
5	Status of this instance	<code>String setStatus(String status)</code> <code>String getStatus()</code>
6	Set the composite instance title	<code>void setCompositeInstanceTitle(String title)</code>
7	Set the indexes that can be used for searching	<code>void setIndex(int i, String value)</code> <code>String getIndex(int i)</code>
8	Priority	<code>void setPriority(int priority)</code> <code>int getPriority()</code>
9	Set and get the creator of the BPEL instance	<code>void setCreator(String creator)</code> <code>String getCreator()</code>
10	Second primary key	<code>void setCustomKey(String customKey)</code> <code>String getCustomKey()</code>
11	Metadata for generating lists	<code>void setMetadata(String metadata)</code> <code>String getMetadata()</code>
12	Access preference	<code>String getPreference(String key)</code>
13	Add an entry to the audit trail	<code>void addAuditTrailEntry(String message, Object detail)</code> <code>void addAuditTrailEntry(Throwable t)</code>

S.No	Features	Function
14	Access and update variables stored in the scope	Object getVariableData(String name) throws BPELFault  Object getVariableData(String name, String part_Or_Query) throws BPELFault  Object getVariableData(String name, String part, String query)
15	Set variable data	void setVariableData(String name, Object value)  void setVariableData(String name, String part, Object value)  void setVariableData(String name, String part, String query, Object value)

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Explain BPEL components, activities, and partner links:** We discussed in detail about the various BPEL activities supported by BPEL 1.1 and BPEL 2.0 template
- **Describe synchronous and asynchronous BPEL processes:** We discussed in detail about the synchronous and asynchronous BPEL processes, its features, and activities associated specifically with these processes
- **Describe message transformations and XSLT:** We discussed about the various techniques available for messages transformations and how XSLT is integrated with Oracle SOA Suite 11g
- **Explain parallel flow and conditional branching:** We discussed in detail about the parallel flow and conditional branching features supported by BPEL through Flow, FlowN, Pick, and ForEach activities
- **Explain BPEL integration with Java:** We discussed in detail about the BPEL features available to integrate with Java.



## Self-review questions

1. Compensation activity can help to roll back a transaction.
  - a. True
  - b. False
2. Checkpoint activity is available to the BPEL process from 10.1.3.3.
  - a. True
  - b. False
3. Parallel processing of a business scenario can be mimicked in BPEL using the \_\_\_\_\_ activity.
  - a. Flow
  - b. Pick
  - c. Scope
  - d. While
4. Binding faults can be caught using.
  - a. CatchAll
  - b. Catch
  - c. Fault Handler Scope
  - d. Main Scope Only
5. The `doTransformation` function takes multiple input parameters for transformation.
  - a. True
  - b. False
6. Events can be generated from BPEL using \_\_\_\_\_.
  - a. EDL file
  - b. Invoke activity
  - c. Pick activity
  - d. Partner Links
7. Pick without Alarm can be the first activity in the BPEL process.
  - a. True
  - b. False

8. Synchronous response time in a BPEL process is controlled by:
  - a. Partner Links used in BPEL
  - b. A Container Property that determines the wait time
  - c. `bpel.xml` property
  - d. It cannot be controlled by the engine
9. BPEL exposed as a service through the SCA services can have multiple operations.
  - a. True
  - b. False
10. BPEL with reply activity is a \_\_\_\_\_ type of BPEL.
  - a. Synchronous
  - b. Asynchronous
  - c. One-Way
  - d. None of the above
11. Faults are propagated to the caller service through:
  - a. Reply activity
  - b. Callback activity
  - c. Throw
  - d. None of the above
12. Process is terminated when a fault is thrown from the BPEL process.
  - a. True
  - b. False
13. Java integration with BPEL is natively supported by BPEL standards.
  - a. True
  - b. False
14. Synchronous BPEL processes will use two different ports to communicate with the caller.
  - a. True
  - b. False

15. The following icon represents which of the following list of options:



- a. Add Entity
- b. Bind Entity
- c. Create Entity
- d. None of the Above

## **Additional reference**

For more details and additional references, please visit the following links:

[http://download.oracle.com/docs/cd/E12839\\_01/integration.1111/e10224/bp\\_appx\\_ref.htm](http://download.oracle.com/docs/cd/E12839_01/integration.1111/e10224/bp_appx_ref.htm)

[http://download.oracle.com/docs/cd/E12839\\_01/integration.1111/e10224/bp\\_parallel.htm](http://download.oracle.com/docs/cd/E12839_01/integration.1111/e10224/bp_parallel.htm)

[http://download.oracle.com/docs/cd/E12839\\_01/integration.1111/e10224/bp\\_java.htm](http://download.oracle.com/docs/cd/E12839_01/integration.1111/e10224/bp_java.htm)

# 6






## Advanced BPEL Concepts

*Chapter 5, Orchestrating Services with BPEL*, introduced BPEL concepts to you. This chapter will provide more insight into some of the advanced features that are available with Oracle BPEL. We will discuss the following list of advanced features supported by Oracle BPEL:

- **Exception handling in composite applications:** In this section we will discuss the exception handling techniques that are available in Oracle BPEL.
- **The fault management framework:** In this section we will discuss the fault management framework provided by Oracle to handle faults in Oracle SCA runtime. This section will also bring out the best practices that need to be followed when defining a fault management.
- **Compensation handling within a BPEL process:** In this section we will discuss the compensation handling techniques that are available in the BPEL process.
- **Correlation concepts:** This section describes the correlation techniques that are available in BPEL.

## Exception handling in composite applications

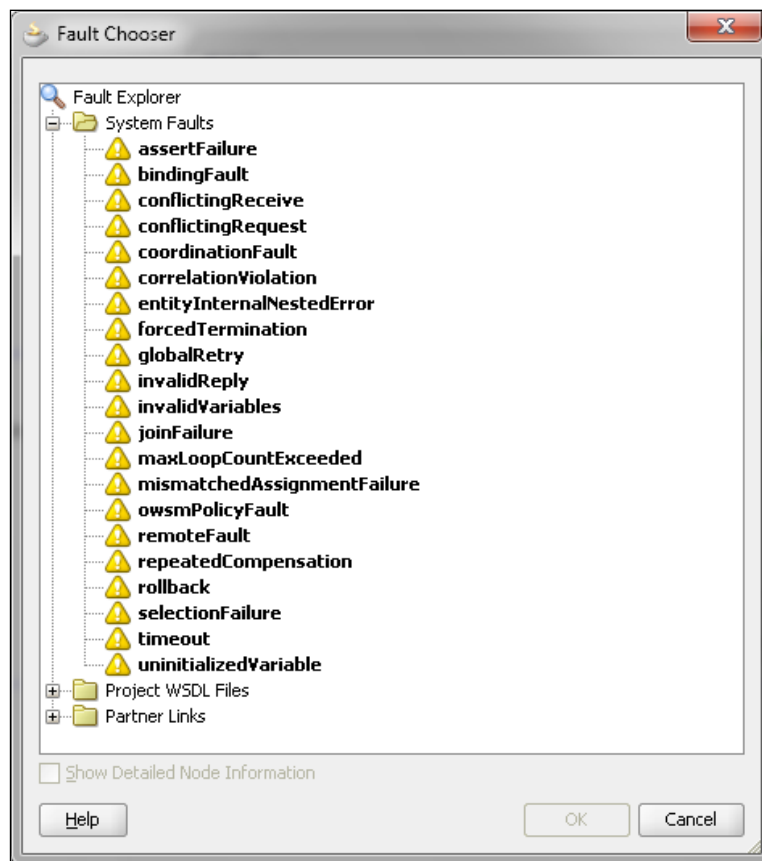
Exception handling is a key component of any computer programming paradigm. Exceptions are raised and propagated as faults in the BPEL context. Error handling in service composite applications can be done using the following list of activities provided by BPEL:

Sr. no.	Activity name	BPEL icon	Description
1	Catch	 Catch	Catch blocks help to catch a process-specific fault type. System/Business faults and fault message types, defined in <b>partner links</b> WSDLs, can be handled using this Catch block.
2	CatchAll	 CatchAll	CatchAll block that are added to main or specific blocks of code can be used to handle any of the faults that are not handled by the Catch blocks.
3	Throw	 ThrowGenericErrorMessage	Throw activity is used to throw the fault/error messages to the BPEL engine to show the process/composite state/status as faulted.
4	Reply	 ReplyFault	Reply activity is used to propagate the fault messages back to a BPEL process.
5	Terminate	 TerminateTheProcess	This activity helps to terminate a BPEL process.

## System faults and custom faults

System faults are propagated by the Oracle SCA infra to BPEL when any system level issues are encountered. The list of system faults that are available with SOA Suite 11.1.1.5 are shown in the next screenshot.

The most commonly occurring system faults are `remoteFault` and `bindingFault`. Rollback fault is the key fault message that is used in BPEL. The rollback fault message is used to rollback a transaction. Faults propagated to the caller will rollback the transaction.



Custom fault messages that are to be used in BPEL are defined through WSDL. Fault messages defined in the WSDL can be used to propagate the faults to the caller, which can be handled in the calling application for error scenarios.

The following listing shows WSDL with a fault message defined in it:

**Listing 6.1**

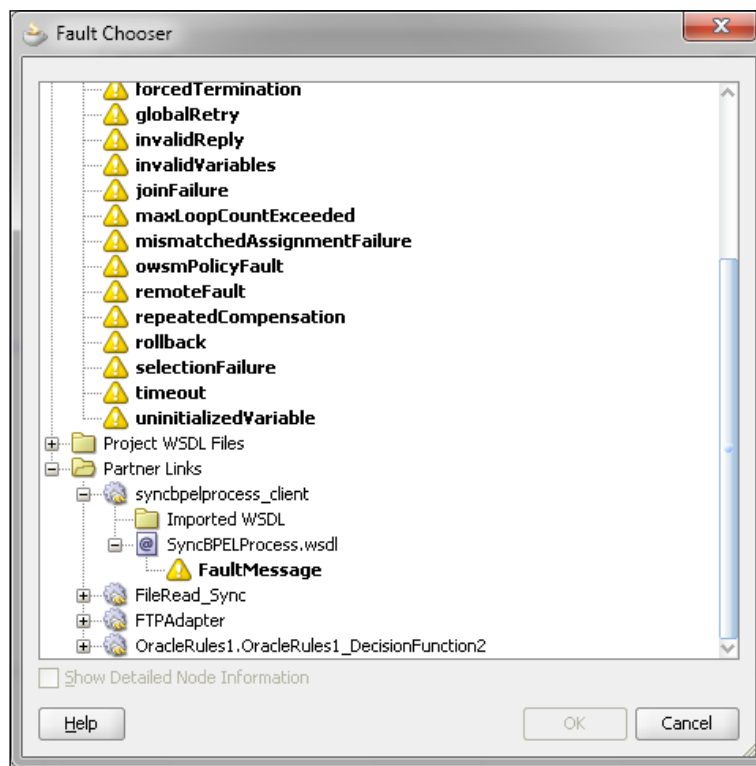
```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="SyncBPELProcess"
targetNamespace="http://xmlns.oracle.com/SOAApplication/SOAPProject
/SyncBPELProcess"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:client="http://xmlns.oracle.com/SOAApplication/SOAPProject
/SyncBPELProcess"
xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/">
<wsdl:types>
  <schema xmlns="http://www.w3.org/2001/XMLSchema">
    <import namespace="http://xmlns.oracle.com/SOAApplication
      /SOAPProject/SyncBPELProcess"
      schemaLocation="xsd/SyncBPELProcess.xsd"/>
  </schema>
</wsdl:types>
<!--
~~~~~
MESSAGE TYPE DEFINITION - Definition of the message types used as
part of the port type definitions
~~~~~
~~~ -->
<wsdl:message name="SyncBPELProcessRequestMessage">
  <wsdl:part name="payload" element="client:process"/>
</wsdl:message>
<wsdl:message name="SyncBPELProcessResponseMessage">
  <wsdl:part name="payload" element="client:processResponse"/>
</wsdl:message>
<wsdl:message name="FaultMessage">
  <wsdl:part name="payload" element="client:faultResponse"/>
</wsdl:message>
<!-- portType implemented by the SyncBPELProcess BPEL process -->
<wsdl:portType name="SyncBPELProcess">
  <wsdl:operation name="process">
    <wsdl:input message="client:SyncBPELProcessRequestMessage"/>
    <wsdl:output message="client:SyncBPELProcessResponseMessage"/>
    <wsdl:fault message="client:FaultMessage"
      name="FaultMessage"/>
  </wsdl:operation>
</wsdl:portType>
<!--
```

```

~~~~~
~~~
PARTNER LINK TYPE DEFINITION
~~~~~

-->
<plnk:partnerLinkType name="SyncBPELProcess">
<plnk:role name="SyncBPELProcessProvider">
<plnk:portType name="client:SyncBPELProcess"/>
</plnk:role>
</plnk:partnerLinkType>
</wsdl:definitions>

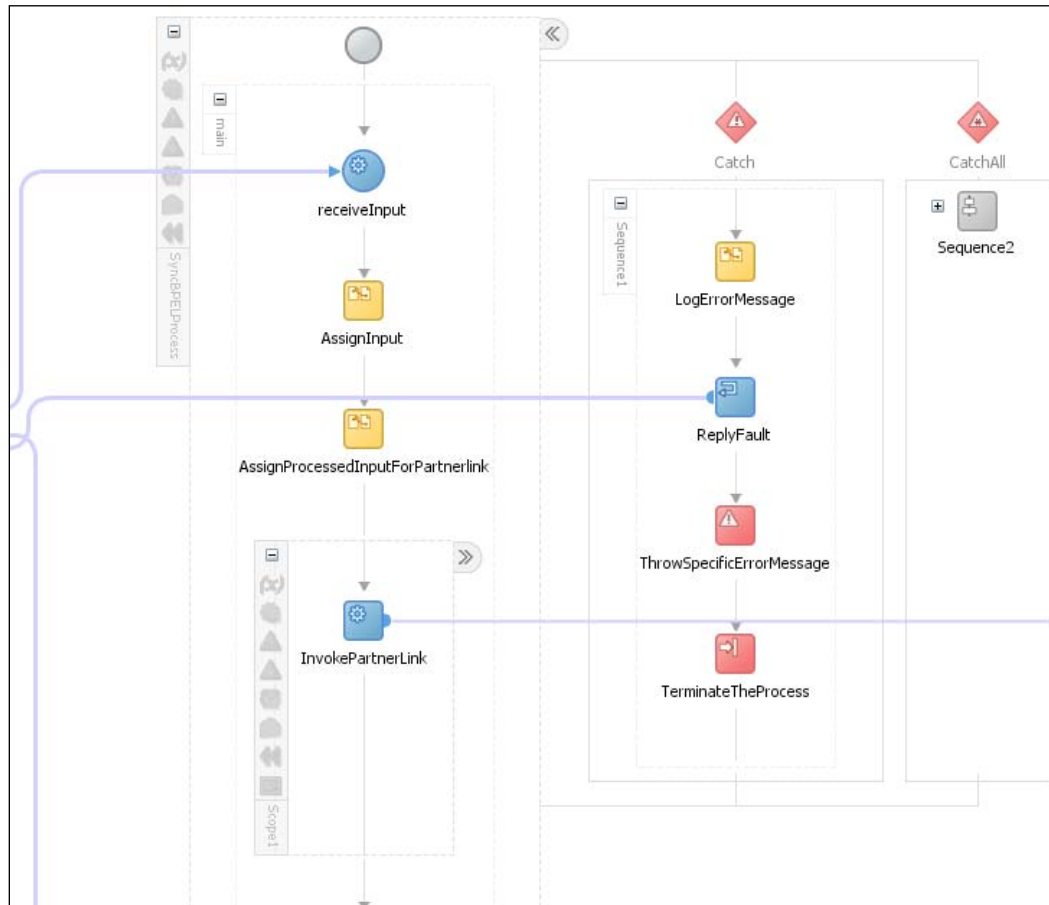
```



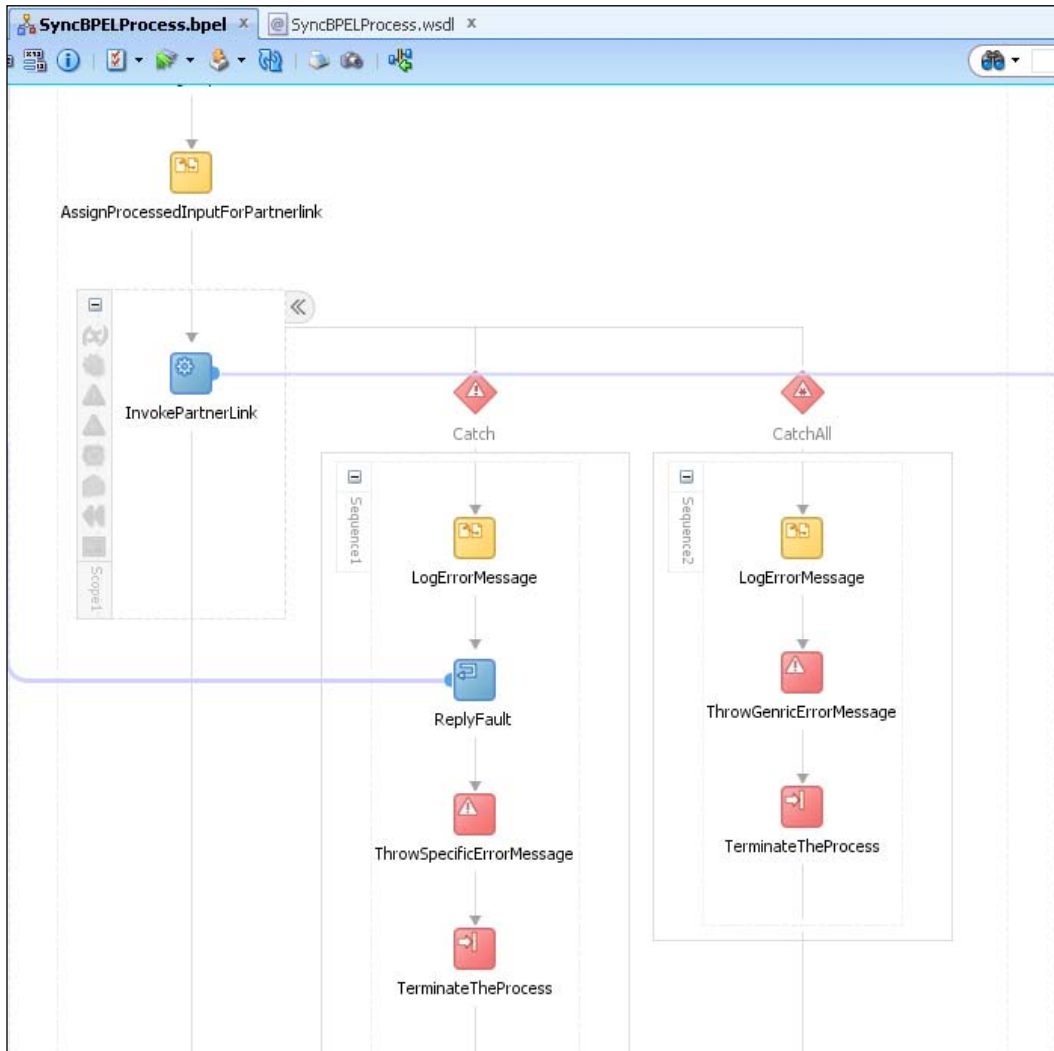
The previous screenshot shows **Fault Chooser** with a custom fault defined in WSDL. This can be used in choose the faults to be handled by the Catch block.



The following screenshot is of the main scope showing Catch and CatchAll activities:




The following screenshot is of the local scope handling `Catch` and `CatchAll` blocks:



## The fault management framework

Fault management is a declarative environment through which the faults can be managed. Both system and business faults can be managed through a fault management framework. The fault management framework provides features to manage both BPEL and Mediator faults. The Oracle SOA Suite 11g fault management framework can be leveraged by writing custom `fault-policies.xml` files to declare possible faults and their associated actions. These policy files can then be associated with composites or components through the `fault-bindings.xml` file.

The following listing shows a sample fault policy with retry options for `remoteFault` and `bindingFault`:

 Fault policies take higher precedence over Catch and CatchAll blocks.

**Listing 6.2:** A fault policy, defined with retry options for `remoteFault` and `bindingFault`, is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<faultPolicies xmlns="http://schemas.oracle.com/bpel/faultpolicy">
  <faultPolicy version="2.0.1" id="SOAFaultPolicy"
    xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:bpelx="http://schemas.oracle.com/bpel/faultpolicy"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Conditions>
      <faultName
        xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
        name="bpelx:remoteFault">
        <condition>
          <action ref="custom-retry"/>
        </condition>
      </faultName>
      <faultName
        xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
        name="bpelx:bindingFault">
        <condition>
          <action ref="custom-retry"/>
        </condition>
      </faultName>
    </Conditions>
    <Actions>
      <Action id="custom-retry">
        <retry>
```

---

```

        <retryCount>2</retryCount>
        <retryInterval>4</retryInterval>
        <retryFailureAction ref="custom-recovery"/>
    </retry>
</Action>
<Action id="custom-recovery">
    <javaAction
        className="custom.soa.extensions.custom.FaultAction"
        defaultAction="ora-human-intervention"
        propertySet="Fault">
        <returnValue value="MANUAL"
            ref="ora-human-intervention"/>
    </javaAction>
</Action>
<Action id="ora-human-intervention">
    <humanIntervention/>
</Action>
</Actions>
<Properties>
    <propertySet name="Fault">
        <property name="EventCode">SOAFaultPolicy</property>
        <property name="ShortDescription">
            Remote, Binding or Mediator Fault Caught in
            SOAFaultPolicy
        </property>
        <property name="LongDescription">
            Remote, Binding or Mediator Fault Caught in
            SOAFaultPolicy
        </property>
        <property name="Severity">30</property>
    </propertySet>
</Properties>
</faultPolicy>
</faultPolicies>

```

The following listing shows a sample fault policy with re-throw options for `remoteFault` and `bindingFault`:

**Listing 6.3:** The following is a fault policy defined with re-throw options for `remoteFault` and `bindingFault`:

```

<faultPolicy version="2.0.1" id="SOAFaultPolicy"
    xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:bpelx="http://schemas.oracle.com/bpel/faultpolicy"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Conditions>

```

```
<faultName
  xmlns:bpelx=http://schemas.oracle.com/bpel/extension
  name="bpelx:remoteFault">
  <condition>
    <action ref="custom-rethrow-action"/>
  </condition>
</faultName>
<faultName
  xmlns:bpelx=http://schemas.oracle.com/bpel/extension
  name="bpelx:bindingFault">
  <condition>
    <action ref="custom-rethrow-action"/>
  </condition>
</faultName>
<faultName
  xmlns:medns="http://schemas.oracle.com/mediator/faults"
  name="medns:mediatorFault">
  <condition>
    <action ref="custom-error-action"/>
  </condition>
</faultName>
</Conditions>
<Actions>
  <Action id="custom-rethrow-action">
    <javaAction
      className="custom.soa.extensions.custom.FaultAction"
      defaultAction="ora-rethrow-fault" propertySet="Fault">
      <returnValue value="OK" ref="ora-rethrow-fault"/>
    </javaAction>
  </Action>
  <Action id="custom-error-action">
    <javaAction
      className=" custom.soa.extensions.custom.FaultAction"
      defaultAction="ora-rethrow-fault" propertySet="Fault">
      <returnValue value="RETHROW" ref="ora-rethrow-fault"/>
    </javaAction>
  </Action>
  <Action id="ora-rethrow-fault">
    <rethrowFault/>
  </Action>
</Actions>
<Properties>
  <propertySet name="Fault">
    <property name="EventCode">SOAFaultPolicy</property>
    <property name="ShortDescription">
      Remote, Binding or Mediator Fault Caught in
      SOAFaultPolicy
    </property>
  </propertySet>
</Properties>
```

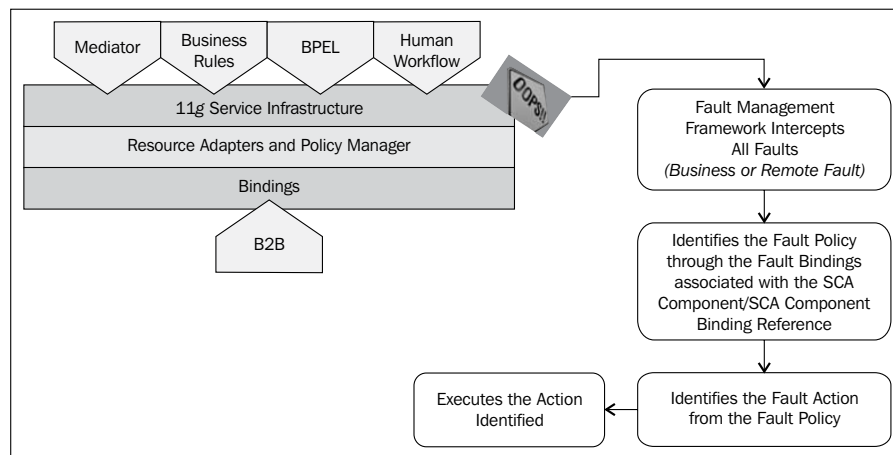
```

    </property>
    <property name="LongDescription">
      Remote, Binding or Mediator Fault Caught in SOAFaultPolicy
    </property>
    <property name="Severity">30</property>
  </propertySet>
</Properties>
</faultPolicy>

```

## The fault policy resolution process

The following diagram helps you to understand the fault policy resolution process involved in identifying and handling faults:



The framework looks for fault policy bindings in the same directory as the `composite.xml` file of the SOA composite application or in a remote location identified by the following two properties:

**Listing 6.4:** The following are `composite.xml` properties to identify fault policies.

```

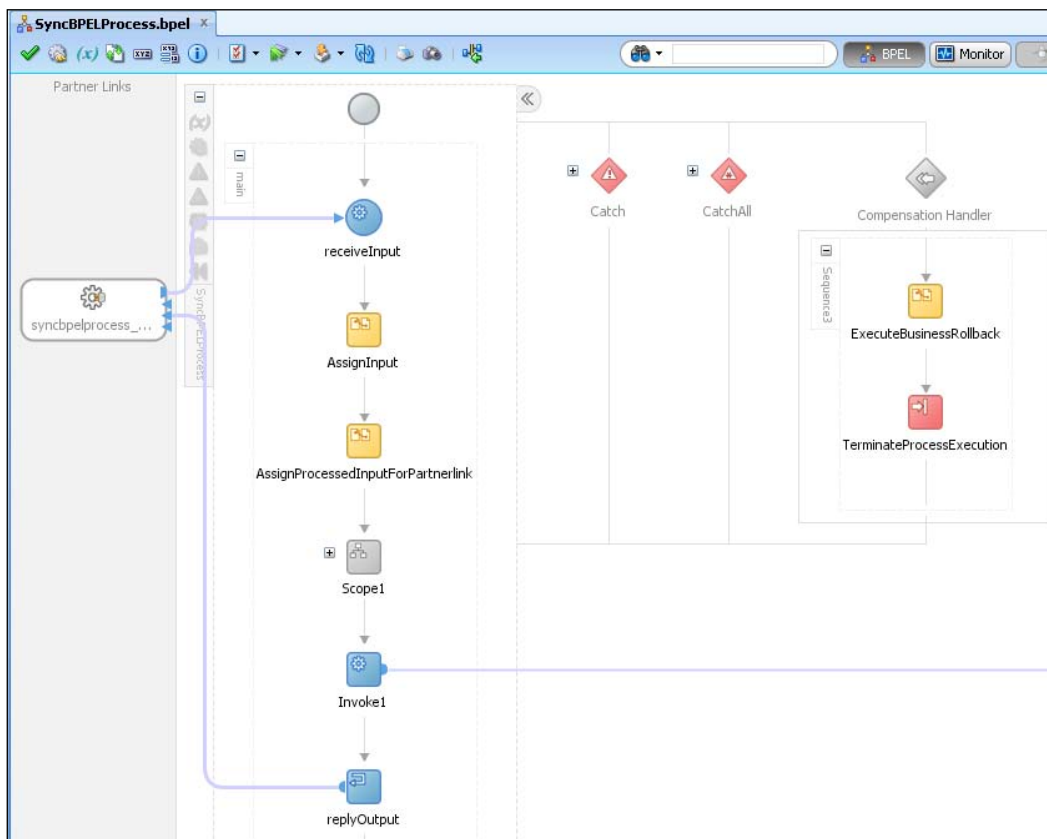
<property name="oracle.composite.faultPolicyFile">
  oramds://apps/faultpolicyfiles/fault-policies.xml
</property>
<property name="oracle.composite.faultBindingFile">
  oramds://apps/faultpolicyfiles/fault-bindings.xml
</property>

```

## Compensation handling within a BPEL process

**Compensation** is a new paradigm in the programming context. This helps to do a business rollback. This is applicable to the idempotent transaction context. Transactions which are committed during the business process execution that require a business rollback can be accomplished using the compensation handler. A compensation handler helps to rollback the transactions that are completed in the previous steps of the BPEL process.

The following screenshot shows the compensation handler in the main scope. A compensation handler can be added to the unit scope level as well.



## Correlation concepts

**Correlation** is a technique that helps to correlate messages between a producer and consumer of the messages in an asynchronous transaction. Correlation helps in scenarios where the interactions are complex. **Mid-process receive** uses correlation sets to correlate messages between incoming message and running instances that are waiting for a message to continue its operations. To understand correlation sets, it is necessary to know about property and property sets. Property and property set information are defined in WSDL.

## Property and property alias

A property is like a variable that stores information which will be used to match the running process instance to a new message arrival.

A **property alias** is used in two situations – as an assign operation to define what information needs to be stored in the property and also as a query to match the information received in a message with the information stored in the property.

**Listing 6.5:** Property and property alias are defined in the process WSDL as follows:

```
<wsdl:definitions name="CorrelationSample"
  targetNamespace="http://xmlns.oracle.com/CorrelationSetApp_jws/
  CorrelationSetPrj/CorrelationSetProcess"
  xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/"
  xmlns:client="http://xmlns.oracle.com/CorrelationSetApp_jws/
  CorrelationSetPrj/CorrelationSetProcess"
  xmlns:pns1="http://xmlns.oracle.com/CorrelationSetApp_jws/
  CorrelationSetPrj/CorrelationSetProcess/correlationset"
  xmlns:bpws="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <plnk:partnerLinkType name="CorrelationSetProcess">
    <plnk:role name="CorrelationSetProcessProvider">
      <plnk:portType name="client:CorrelationSetProcess"/>
    </plnk:role>
    <plnk:role name="CorrelationSetProcessRequester">
      <plnk:portType name="client:CorrelationSetProcessCallback"/>
    </plnk:role>
  </plnk:partnerLinkType>
  <bpws:propertyAlias propertyName="pns1:matchValue_prop"
    messageType="client:CorrelationSetProcessRequestMessage"
    part="payload" query="/client:process/client:input"/>
  <bpws:propertyAlias propertyName="pns1:matchValue_prop"
    messageType="client:CorrelationSetIntermediateRequestMessage"
    part="payload" query="/client:process/client:input"/>
```



**Listing 6.6** – The correlation set definition in the BPEL process is shown in the below following sample:

```
<process name="CorrelationSampleProcess"
targetNamespace="http://xmlns.oracle.com/CorrelationSetApp_jws/Cor
relationSetPrj/CorrelationSetProcess"
xmlns="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
xmlns:client="http://xmlns.oracle.com/CorrelationSetApp_jws/Correl
ationSetPrj/CorrelationSetProcess"
xmlns:ora="http://schemas.oracle.com/xpath/extension"
xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
xmlns:bpws="http://schemas.xmlsoap.org/ws/2003/03/business-
process/"
xmlns:ns1="http://xmlns.oracle.com/CorrelationSetApp_jws/Correlati
onSetPrj/CorrelationSetProcess/correlationset">
  <partnerLinks>
    <!--
The 'client' role represents the requester of this service. It is
used for callback. The location and correlation information
associated with the client role are automatically set using WS-
Addressing.
-->
    <partnerLink name="correlationsetprocess_client"
      partnerLinkType="client:CorrelationSetProcess"
      myRole="CorrelationSetProcessProvider"
      partnerRole="CorrelationSetProcessRequester"/>
  </partnerLinks>
  <variables>
    <!-- Reference to the message passed as input during initiation -->
    <variable name="inputVariable"
      messageType="client:CorrelationSetProcessRequestMessage"/>
    <!-- Reference to the message that will be sent back to the
requester during callback -->
    <variable name="outputVariable"
      messageType="client:CorrelationSetProcessResponseMessage"/>
    <variable
name="receiveIntermediateOperation_intermediateOperation_
InputVariable"
messageType="client:CorrelationSetIntermediateRequestMessage"/>
  </variables>
  <correlationSets>
    <correlationSet name="MyCorrelationSet"
      properties="ns1:matchValue_prop"/>
  </correlationSets>
```

---

```

<sequence name="main">
  <!-- Receive input from requestor. (Note: This maps to operation
  defined in CorrelationSample.wsdl) -->
  <receive name="receiveInput"
    partnerLink="correlationsetprocess_client"
    portType="client:CorrelationSetProcess"
    operation="process" variable="inputVariable"
    createInstance="yes">
    <correlations>
      <correlation initiate="yes" set="MyCorrelationSet"/>
    </correlations>
  </receive>
  <!-- Asynchronous callback to the requester. (Note: the callback
  location and correlation id is transparently handled using WS-
  addressing.) -->
  <scope name="AnyActivity">
    <empty name="Empty_1"/>
  </scope>
  <receive name="receiveIntermediateOperation"
    createInstance="no"
    variable="receiveIntermediateOperation_intermediateOperation_
    InputVariable"
    partnerLink="correlationsetprocess_client"
    portType="client:CorrelationSetProcess"
    operation="intermediateOperation">
    <correlations>
      <correlation initiate="no" set="MyCorrelationSet"/>
    </correlations>
  </receive>
  <scope name="AnyActivity2">
    <empty name="Empty_2"/>
  </scope>
  <invoke name="callbackClient"
    partnerLink="correlationsetprocess_client"
    portType="client:CorrelationSetProcessCallback"
    operation="processResponse" inputVariable="outputVariable"/>
</sequence>
</process>

```

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Describe exception handling in composite applications:** We discussed about the fault/exception handling techniques available in composite applications. We also identified the difference between system and custom faults.
- **Describe the fault management framework:** We discussed in detail about the various features provided by Oracle SOA Suite in handling faults (BPEL and Mediator) declaratively through the fault management framework.
- **Describe compensation handling within a BPEL process:** We have discussed in detail compensation activity and its significance.
- **Describe correlation concepts:** We discussed in detail about the significance of correlations in asynchronous process interactions.

## Self-review questions

1. Multiple `CatchAll` blocks can be used in a BPEL.
  - a. True
  - b. False
2. Multiple `CatchAll` blocks can be defined per BPEL scope.
  - a. True
  - b. False
3. A fault that occurred in a composite can be handled using \_\_\_\_\_.
  - a. Fault management framework
  - b. BPEL error process
  - c. BPEL fault process
  - d. `Catch` and `CatchAll` blocks
4. Custom Java actions are available to handle the fault through the fault management framework.
  - a. True
  - b. False

5. Fault policies need not be explicitly specified in the composite definition.
  - a. True
  - b. False
6. Fault policies take higher precedence over the locally defaulted fault handling blocks in the SOA composites.
  - a. True
  - b. False
7. Fault policies can vary for each and every composite.
  - a. True
  - b. False
8. Sync fault policies can have retry options for faulted messages.
  - a. True
  - b. False
9. Async fault policies can have retry options for faulted messages.
  - a. True
  - b. False
10. Fault policies can be used only for the system faults.
  - a. True
  - b. False
11. Multiple compensation handlers can be defined per BPEL.
  - a. True
  - b. False
12. Multiple compensation handlers can be defined per BPEL scope.
  - a. True
  - b. False
13. The property alias required for correlation is defined in the BPEL file.
  - a. True
  - b. False

14. Correlation sets are defined in the BPEL file.
  - a. True
  - b. False
15. The `getFaultAsString()` function can be used in both `Catch` and `CatchAll` blocks.
  - a. True
  - b. False

# 7

## Working with Mediator Concepts and Features

Oracle Mediator provides a lightweight framework to mediate data between various components and references within a composite application. Mediator converts messages and protocol to facilitate communication between different interfaces exposed by different components, which are wired together to build a SOA composite application. The Mediator component can be invoked by business event, adapter event, or service call. A Mediator component can evaluate routing rules, perform transformations, validate, and either invoke services or raise business events.

The following illustration would help you to understand the Mediator functionalities. A Mediator can accept data contained in a text file from an application or service, transform it to a format appropriate for updating a database that serves as a customer repository, and then route and deliver the data to that database. Oracle Mediator facilitates integration between events and services where service invocations and events can be mixed and matched.

### What the Mediator is not?

The Mediator is an intra-composite mediation component that is deployed within an application. It is responsible for establishing communications between components within a composite, enabling transformation, routing, event delivery, and payload validation inside the composite. The developers use Mediator for assembling multiple components into a composite application which can be deployed, versioned, and managed as a single deployment unit. Oracle ESB can be migrated as Oracle Mediator in 11g, but it will lose its dynamic runtime change capabilities.

The following list of ESB features will not be available with Mediator. Oracle Service Bus complements these features and this will be used as a strategic ESB product going forward.

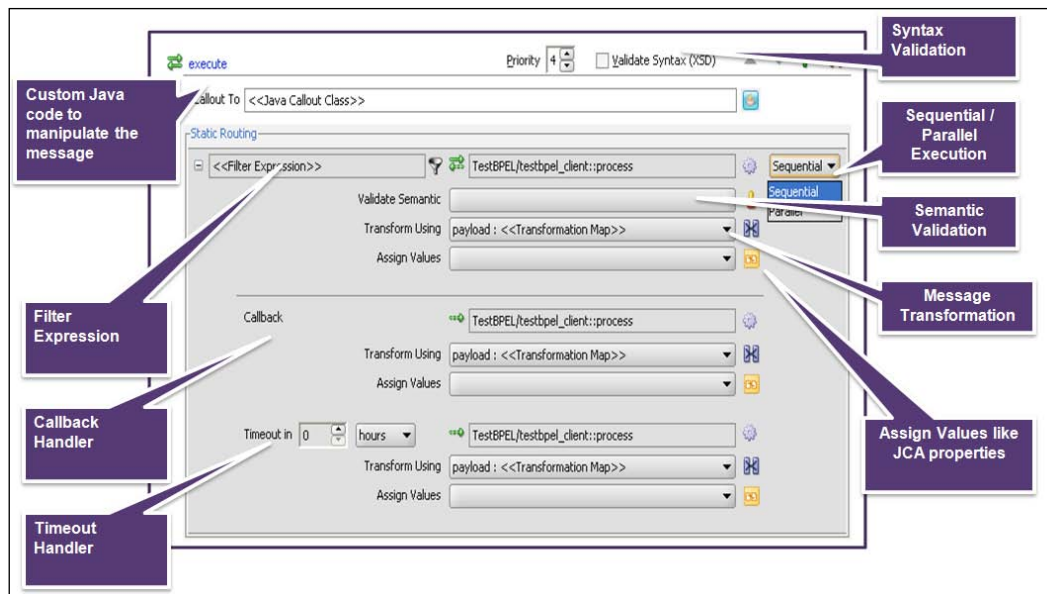
- Virtualized endpoint management
- Configuration framework
- Advanced mediation patterns, such as split-join

This chapter will cover the following exam objectives:

- Explain the Mediator component and its features
- Create and configure a Mediator service component
- Create Mediator routing rules
- Fault Handling in Mediator

## The Mediator component and its features

The Mediator component can be used to mediate the service data between different components in the SCA composites. It can also be used to mediate the service between different services using filtering rules and transformation rules defined at the Mediator.



The following table shows the list of features supported by the Mediator:

S.No	Features	Description
1	EDN Support	Ability to generate and subscribe to events
2	Schematron Validation	Ability to validate the data using Schematron files
3	XPath-based Assignment	Ability to assign values to data in the Mediator
4	Mediator Echo Support	Ability to echo the message using the default echo operation
5	Multiple Part Message Support	Ability to process multipart messages
6	Content-Based and Header-Based Routing	Ability to route data based on the data content or header
7	Sequential and Parallel Routing of Messages	Ability to route the messages in parallel or in sequence
8	Transformations	Ability to transform the messages using XSL
9	Dynamic Routing	Ability to use Business Rules to define the routing rules. The routing information can be changed dynamically using SOA Composer
10	Java callout	Ability to execute a Java code
11	Consistent Fault Policy Framework with BPEL	Ability to handle Mediator errors through Fault Management Framework
12	Synchronous and Asynchronous Interactions	Ability to perform synchronous and asynchronous interaction from the Mediator



## Creating and configuring a Mediator service component and creating Mediator routing rules

Mediator can be added to the SCA Composite as a Component. This can be accomplished by carrying out the following steps:

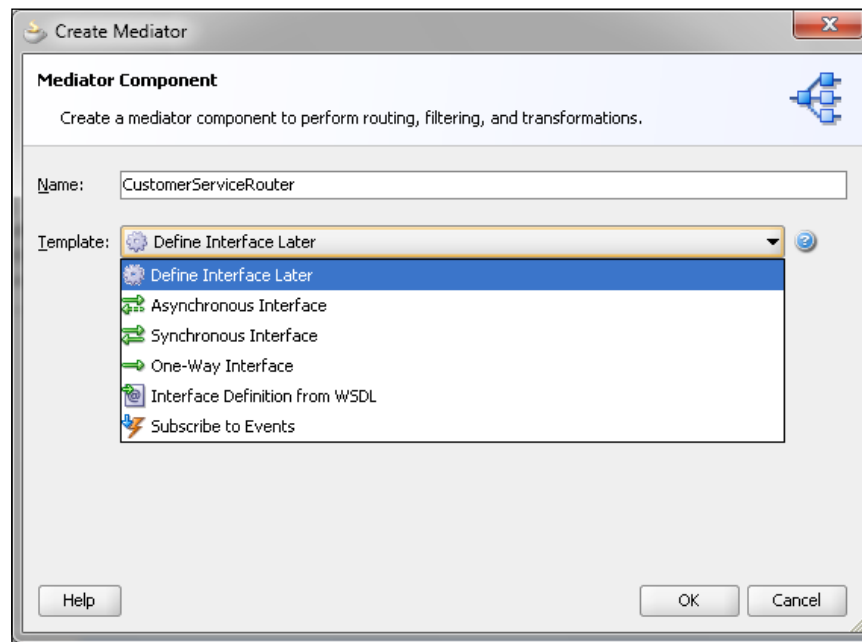
1. Open the `Composite.xml` file.
2. Drag a Mediator **Component Palette** into the **Components** column.
3. In the **Create Mediator** window, enter the component name and select the template.
  - **Define interface Later**
  - **Asynchronous Interface**
  - **Synchronous Interface**
  - **One-Way Interface**
  - **Interface Definition from WSDL**
  - **Subscribe to Events**

The following table shows the description of the templates for each of the Mediator types:

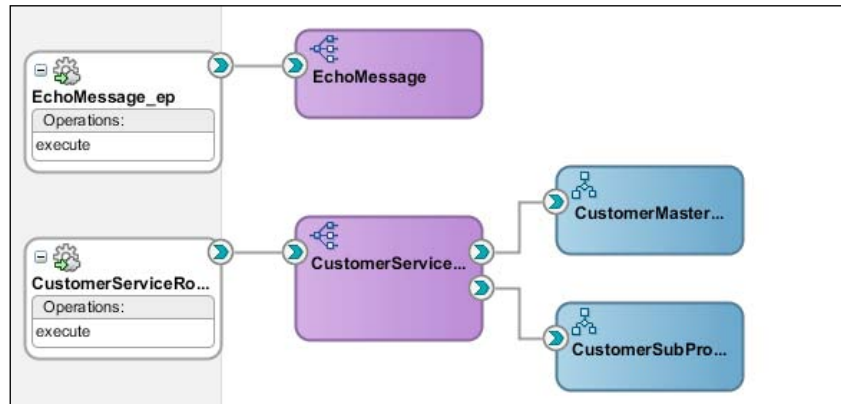
S.No	Template Name	Template Description
1	<b>Define Interface Later</b>	This template allows the user to define the interface for the Mediator later.
2	<b>Synchronous Interface</b>	This template is used to define the synchronous interface definition for the Mediator. In a synchronous interaction, a client sends a request to a service and receives an immediate response.
3	<b>Asynchronous Interface</b>	This template is used to define the asynchronous interface definition for the Mediator. In an asynchronous interaction, a client sends a request to a service but does not block and wait for a reply.
4	<b>One-Way Interface</b>	This template is used to define the one-way interface definition for the Mediator. In a one-way interaction, the client sends a message to the service and the service does not need to reply.

S.No	Template Name	Template Description
5	<b>Interface Definition from WSDL</b>	This template is used to define the interface definition for the Mediator using a predefined WSDL definition. In an interface definition from WSDL, the WSDL file describes the interface of a Mediator such as schemas and operations.
6	<b>Subscribe to Events</b>	This template is used to define the event based subscription for the Mediator.

The following screenshot shows the list of **Mediator Component** templates that are available for the user to configure the Mediator:

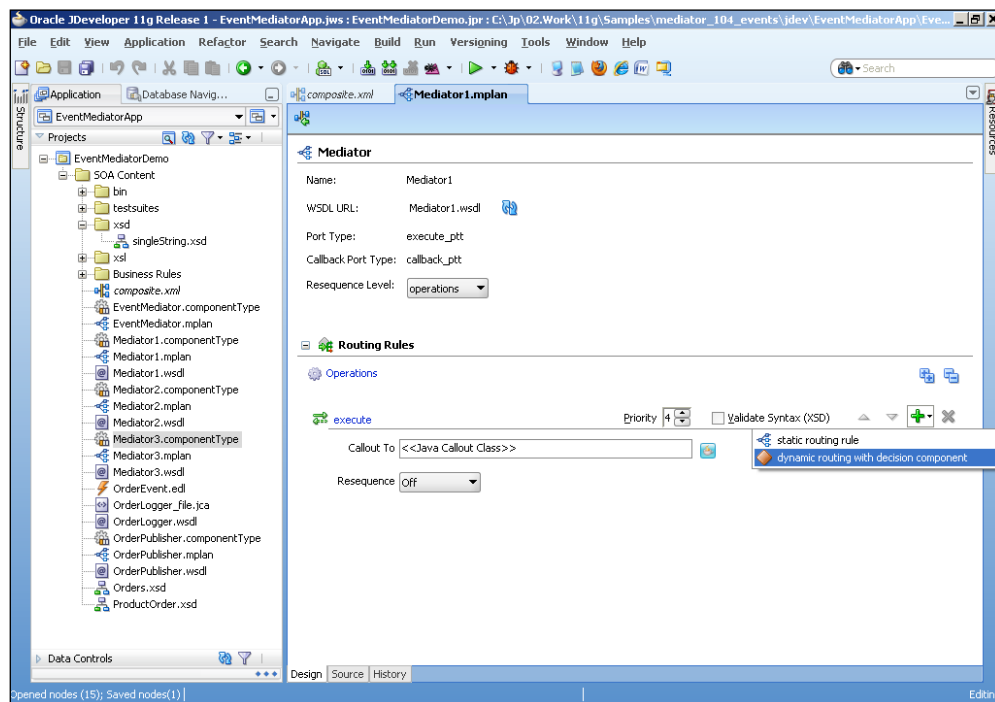


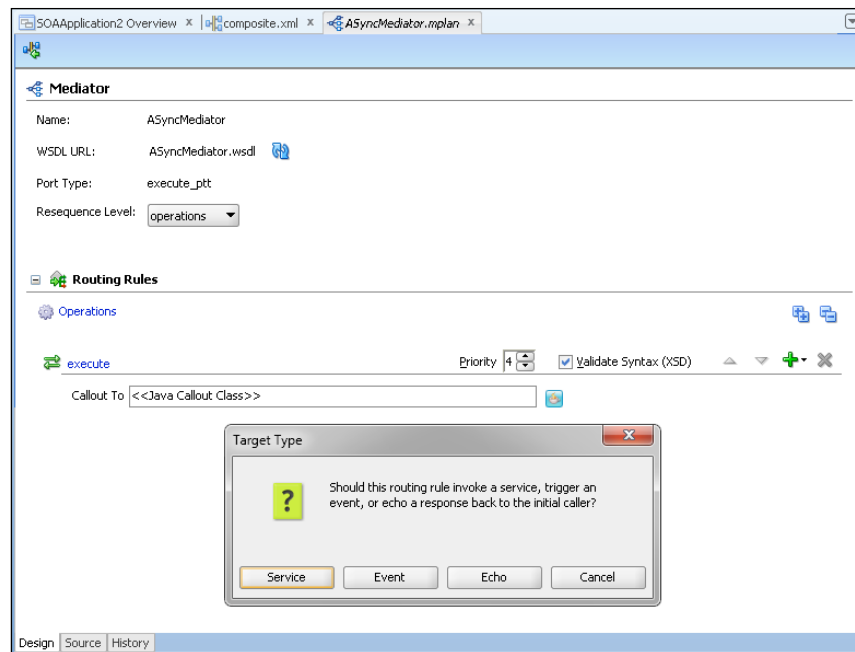
The following screenshot shows the Mediator View in the **Composite** pane:



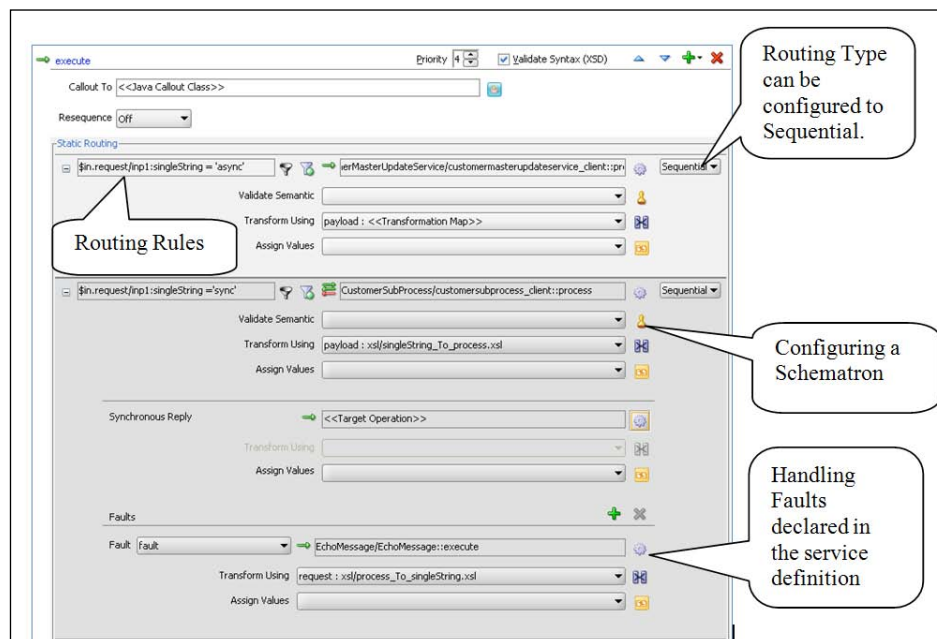
## Configuring a Mediator

The routing rules can be added to a Mediator using the + in the execute line. This allows us to add routing rules to route the message to a service/event or echo the message back to the caller. The routing of the messages can also be based on the preconfigured rules through Business Rules.



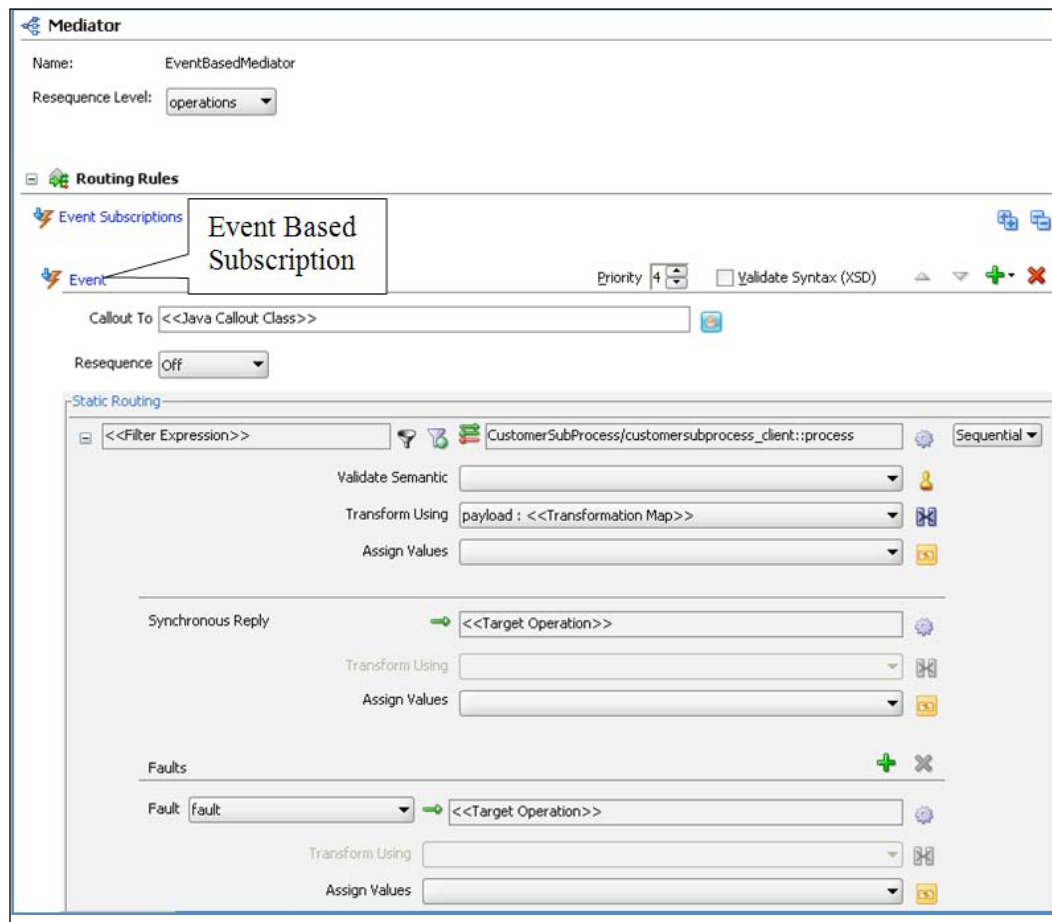


**Configuring a Mediator with target type as Service option:** The following Mediator configuration shows the configuration of the Mediator using service operation calls.



## Configuring a Mediator to subscribe to an Event

The following screenshot of the Mediator configuration shows the configuration using Events. This Mediator will execute the routing rules based on the events received through the EDN layer. The following screenshot illustrates the Mediator subscribing to an event and calling a synchronous service:



## Fault Management in a Mediator

Faults that arise in a Mediator can be handled using the Fault Policy Framework. Refer to the previous chapter (*Chapter 6, Advanced BPEL Concepts*) to understand, in detail, the Fault Policy Framework. Mediator faults are identified using the `http://schema.oracle.com/mediator/faults` namespace and the fault name as `mediatorFault` from the same namespace.

The following XML snippet shows the sample fault policy for handling Mediator faults:

```
<?xml version="1.0" encoding="UTF-8"?>
<faultPolicies>
  <faultPolicy version="1.0" id="MediatorFaultPolicies">
    <Conditions>
      <faultName xmlns:medns="http://schemas.oracle.com/mediator/
faults"
name="medns:mediatorFault">
        <condition>
          <test>contains($fault.mediatorErrorCode, "TYPE_ALL")</test>
          <action ref="ora-retry"/>
        </condition>
      </faultName>
    </Conditions>
    <Actions>
      <Action id="ora-retry">
        <retry>
          <retryCount>3</retryCount>
          <retryInterval>2</retryInterval>
          <exponentialBackoff/>
          <retryFailureAction ref="ora-java"/>
          <retrySuccessAction ref="ora-terminate"/>
        </retry>
      </Action>
    </Actions>
  </faultPolicy>
</faultPolicies>
```

## Mediator error groups

There are five error groups in the Mediator: TYPE\_DATA, TYPE\_METADATA, TYPE\_FATAL, TYPE\_TRANSIENT, TYPE\_INTERNAL, and TYPE\_ALL includes all Mediator fault types.

S.No	Error Group	Error Code	Description
1	TYPE_DATA	TYPE_DATA_ASSIGN	Errors related to data assignment can be identified using this error code.
2	TYPE_DATA	TYPE_DATA_FILTERING	Errors related to payload data filtering in the Mediator can be identified using this error code.
3	TYPE_DATA	TYPE_DATA_TRANSFORMATION	Errors related to payload data transformation in the Mediator can be identified using this error code.
4	TYPE_DATA	TYPE_DATA_VALIDATION	Errors related to payload validation in the Mediator can be identified using this error code.
5	TYPE_METADATA	TYPE_METADATA_FILTERING	Errors related to metadata filtering in the Mediator can be identified using this error code.
6	TYPE_METADATA	TYPE_METADATA_TRANSFORMATION	Errors related to metadata transformation in the Mediator can be identified using this error code.
7	TYPE_METADATA	TYPE_METADATA_VALIDATION	Errors related to validation in the Mediator can be identified using this error code.
8	TYPE_METADATA	TYPE_METADATA_COMMON	Other common errors related to handling metadata can be identified using this error code.
9	TYPE_FATAL	TYPE_FATAL	Errors that can be recovered, and the fault can be identified using these error codes specified for their own specific needs.
10	TYPE_FATAL	TYPE_FATAL_DB	
11	TYPE_FATAL	TYPE_FATAL_CACHE	
12	TYPE_FATAL	TYPE_FATAL_ERRORHANDLING	
13	TYPE_FATAL	TYPE_FATAL_MESSAGING	
14	TYPE_FATAL	TYPE_FATAL_TRANSFORMATION	

S.No	Error Group	Error Code	Description
15	TYPE_TRANSIENT	TYPE_TRANSIENT_MESH	Error-related Service Infrastructure can be identified using this error code.
16	TYPE_TRANSIENT	TYPE_TRANSIENT_MESSAGING	Errors related to JMS such as enqueue and dequeue can be identified using this error code.
17	TYPE_INTERNAL		All errors related to the Mediator internals can be identified using this error code.
18	TYPE_ALL		Above-specified errors are grouped under this error code.

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Explain the Mediator component and its features:** We understood the concepts and features of Oracle Mediator
- **Create and configure a Mediator service component:** We identified the various configuration features provide in Mediator service components and steps required to achieve them
- **Create Mediator routing rules:** This section introduced various routing rules and options available to route the service messages in an SCA assembly

## Self-review questions

1. Validation of XML data can be done in the Mediator.
  - a. True
  - b. False
2. Virtualization Endpoint Management can be achieved using the Mediator.
  - a. True
  - b. False
3. Events can be published from the Mediator.
  - a. True
  - b. False



4. Business Rules can be used to route messages in Mediator.
  - a. True
  - b. False
5. Java Callout is executed before the routing rules.
  - a. True
  - b. False
6. Dynamic routing of messages is possible through \_\_\_\_\_ Mediator.
  - a. One-Way Interface
  - b. Synchronous Interface
  - c. Asynchronous Interface
  - d. All the above
7. Echo option is supported in which of the following patterns in the Mediator?
  - a. Request/Reply
  - b. Request/Reply/Fault
  - c. Request/Callback
  - d. All the above
8. Header properties in the Mediator are automatically propagated to the target service.
  - a. True
  - b. False
9. Which is the right execution order for a given operation in the Mediator?
  - a. 1. Switch - Case 2. Invoke 3. Assign 4. Transform
  - b. 1. Switch - Case 2. Transform 3. Assign 4. Invoke
  - c. 1. Switch - Case 2. Assign 3. Transform 4. Invoke
  - d. 1. Switch - Case 2. Assign 3. Invoke 4. Transform
10. A single Mediator can host multi-default routing rules.
  - a. True
  - b. False

11. If an underlying WSDL definition for the Mediator is changed, it is required to rebuild the Mediator.
  - a. True
  - b. False
12. A Mediator initiated through events has the facility to echo the message back to the caller.
  - a. True
  - b. False



# 8

## Human Workflow

**Workflow** is an essential component in all business process. The mantras for Fusion Applications and SOA are business process orientated and move away from IT-centric or IT-driven processes. The marriage between technology and business using service-orientation application design is bridged by the **Oracle Human Workflow** component.

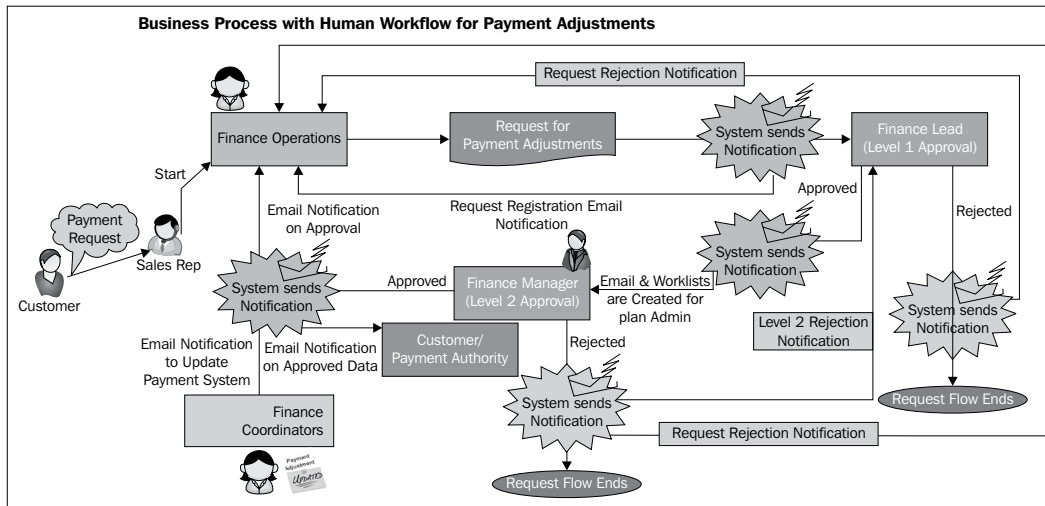
Oracle Human Workflow helps to accomplish automated notification on the process movement or transition between different phases of business, and provides the necessary frameworks and tools to ensure right checks and balances in the business process. Wider adoption of SOA requires centralized workflow management tools. Oracle SOA Suite provides this capability to extend the core business process execution (orchestration) framework to include the human component in the business process automation. Though workflow has been a well-known technology for many years and has been adopted widely in different business applications like Oracle E-Business Suite, PeopleSoft, Siebel, and so on, due to a change in the fundamental approach towards Fusion Application and SOA, it calls for a centralized workflow management tool.

Oracle Human Workflow architecture provides the following features:

- Ability for human beings to interact with business processes that include assignment and routing of tasks to the right users or user group
- Ability to generate notifications related to deadlines, escalations, and other generic normal notifications that would aid in the timely execution of the tasks associated with the business process
- Ability to view the task online and provide approvals or reject the requests generated through business process automation

- Ability to organize, filter, and prioritize the tasks of the end users to productively perform their task
- Ability to manage the vacation and other features required by supervisors and business owners to manage the performance of tasks

The following diagram shows an example of a workflow that involves the human component to accomplish the business processes. The diagram depicts the payment adjustment request scenario:



Oracle Human Workflow can be utilized for automating the process shown in the previous diagram; complete the business process execution from the centralized Oracle SOA Suite engine and it can be externalized from the application layers.

This chapter will cover the following list of exam objectives:

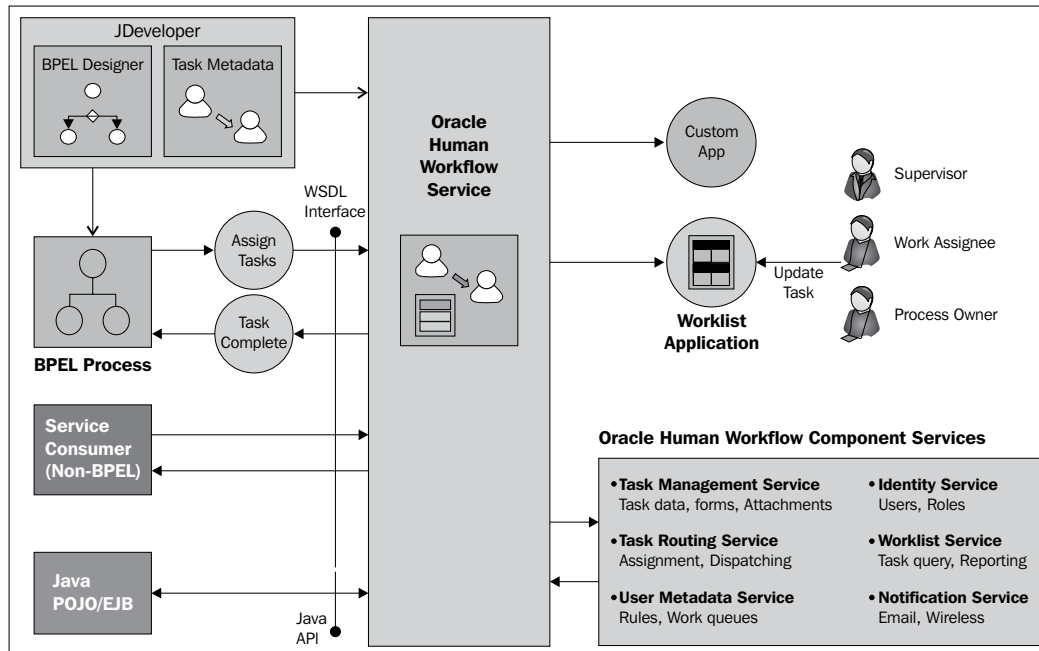
- Describing the Human Workflow architecture, features, and concepts
- Designing human tasks and services
- Invoking a human task from a BPEL process

## Oracle Human Workflow architecture

Oracle Human Workflow architecture include the following components:

- Human Workflow service engine
- Human Workflow services
- Oracle Worklist Application

The following diagram shows the high-level Oracle Human Workflow architecture:



## Human Workflow service engine

Human Workflow service engine is no longer part of the BPEL engine. This is accessible and available as a separate service engine. The BPEL engine interacts with the Human Workflow engine through the services exposed by the Human Workflow service engine and the metadata information that is stored in the MDS about the human task definition.

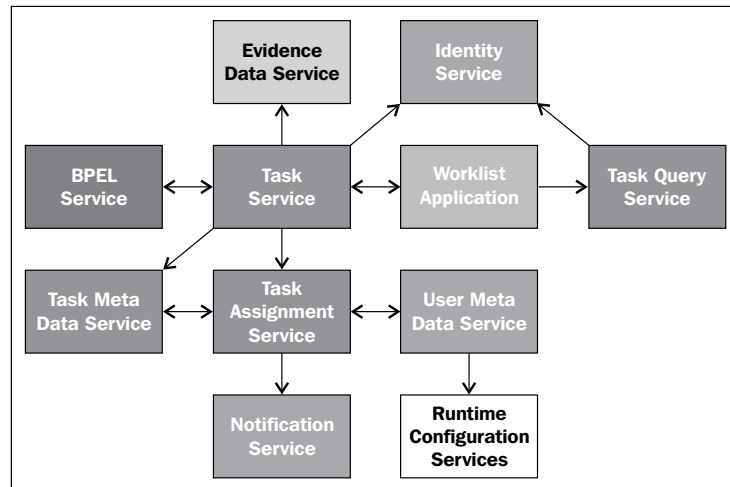
## Human Workflow services

Oracle Human Workflow services includes a list of sub-services that are listed in the following table:

Sr. no.	Service name	Service description
1	Task metadata service	Exposes metadata information related to a task data
2	User metadata service	Manages metadata related to the Human Workflow
3	Task service	Provides task persistence information
4	Task assignment service	Offers services to route, escalate, and assign task information

Sr. no.	Service name	Service description
5	Task query service	Queries a task for a user based on keyword, category, and status
6	Notification service	Sends notification to users using a unified messaging platform
7	Identity service	Provides user, group, and role lookup services
8	Evidence store service	Captures digital signatures and digitally signed tasks
9	Runtime config service	Runtime metadata information of the task service can be managed using this service
10	Task report service	Provides workflow report details

The following flow diagram explains the interaction between the BPEL component and the workflow services interaction:

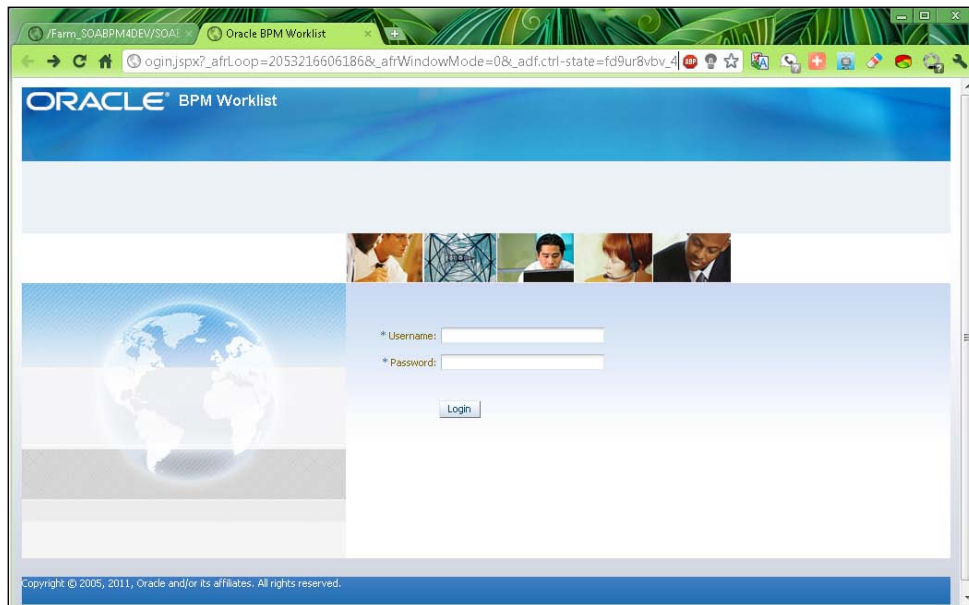


## Oracle Worklist Application

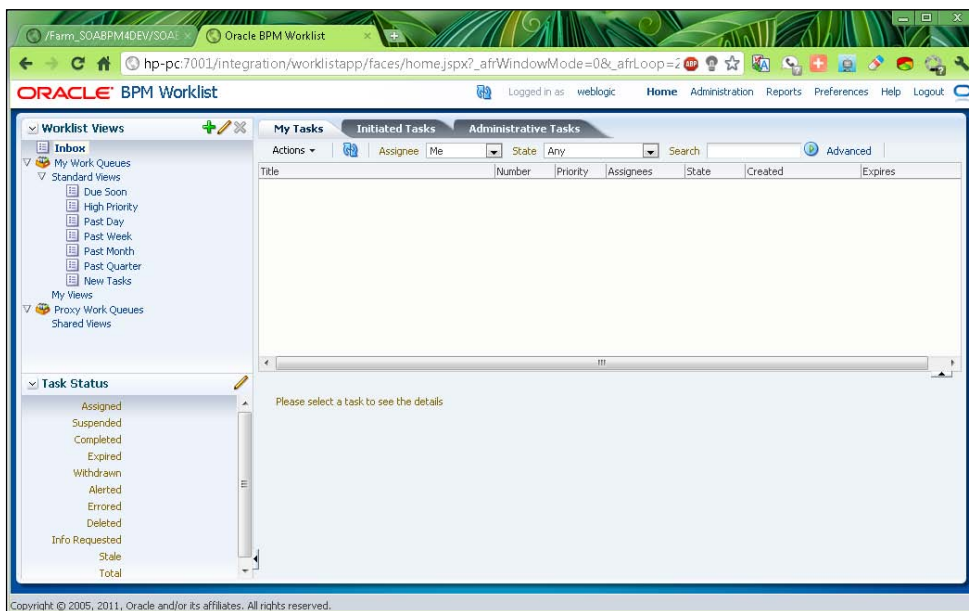
The main component of the Oracle Human Workflow framework is the **Oracle Worklist** Application which is bundled along with Oracle SOA Suite, and it is deployed in the same webLogic domain as Oracle SOA Suite.

The Oracle Worklist Application can be accessed using the following link:

`http://<hostname>:<default-soa-em-port>/integration/worklistapp`



The following screenshot is the landing screen of the Worklist Application:



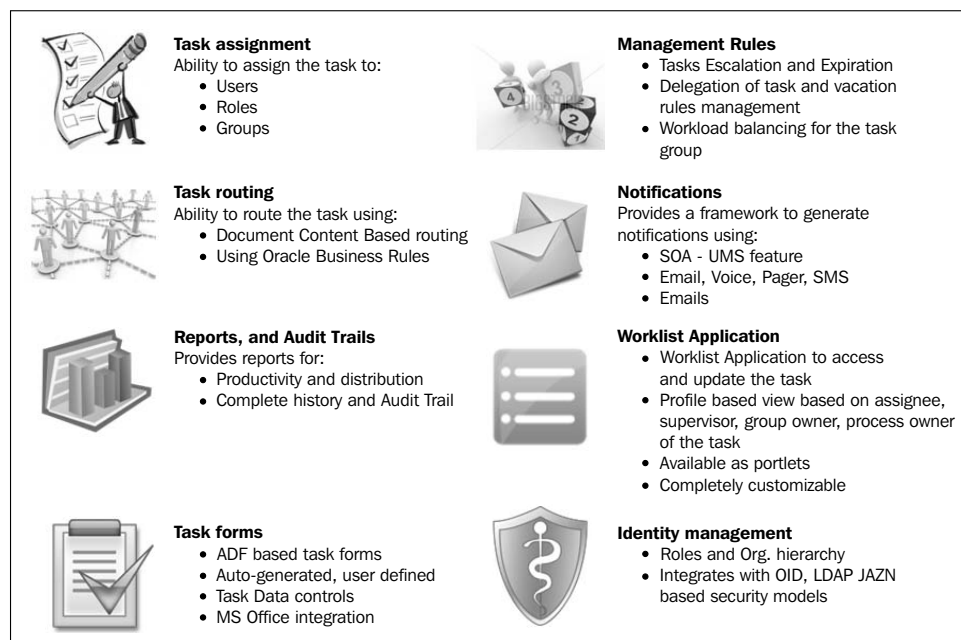


## Oracle Human Workflow features

Oracle Human Workflow provides the following features as part of the Oracle SOA Suite 11g framework:

- Task assignment
- Task routing
- Management rules
- Notifications
- Worklist Application
- Task forms
- Reports and audit trails
- Identity management

The following diagram shows a detailed view of the features provided by the Oracle Human Workflow components:



## Oracle Human Workflow concepts

This section will highlight the following important concepts and terms to be understood in regards to the Oracle Human Workflow:

- Participant
- Task assignment
- Routing policy
  - Static and dynamic routing
  - Rule-based routing
  - Outcome-based routing
- Ad-hoc routing
- Task stakeholders
- Task deadlines
- Notifications
- Task forms
- Stages
- Access rules
- Standalone human tasks
- Callbacks
- Events

### Participant

A participant is the user, group, or application role defined in the task flow definition.

### Task assignment

The task information is the input to the human task on which the task routing policies will be applied to accomplish the desired workflow. A task can be assigned to a user, group, or application roles. Assignment can be static, dynamic, or rules-based assignment. Static assignment always assigns the task to a specific user, or group and value of the assignee is not determined at runtime. Dynamic assignment can be done using the support of identity management APIs that are available along with the SOA Suite. Identity management APIs will allow the clients to lookup on the configured identity store and retrieve the details of the assigned participants.

## Routing policy

The Routing policy is the definition of the task flow that defines the workflow path. This routing policy could be static, dynamic, or a business rule-based definition or it could be a combination of any of them. **Oracle Business Rules** can be used to alter the flow of the workflow based on the content of the data or based on the response from the actors in the workflow. This helps to achieve a dynamic definition of the workflow and the outcome of the business process flow.

## Ad-hoc routing

Ad-hoc routing is an advanced concept in the Oracle Human Workflow where the participant can invite other participants to participate in the workflow to achieve a decision based on the complexity of the decision point to be made.

## Task stakeholders

Oracle Human Workflow framework provides features besides the participant to be stakeholders in the Human Workflow. The following list shows the additional stakeholders provided besides the participants:

- Admins
- Approvers
- Creator
- Owner
- Reviewers

The Task definition provides the feature to specify the access levels to the stakeholders, both at content level as well as at action level. Oracle Human Workflow currently supports the following actions:

- Approve
- Reject
- Acquire
- Ad-hoc route
- Delegate
- Delete
- Escalate
- Information request

- Override routing slip
- Purge
- Push back
- Reassign
- Release
- Renew
- Resume
- Skip current assignment
- Suspend

## **Task deadlines**

The task deadlines action is the action that can be performed on the task if no action is taken for a specified period of time.

- Reminder
- Escalation
- Expiration
- Renewal

## **Notifications**

Notifications are the key part of any workflow technique. Oracle Human Workflow supports the features to send a notification to the required user or a group through the following list of channels:

- E-mail
- Voice
- Instance messaging
- Short Message Service

## **Task forms**

Task forms are the way by which users will interact with a task based on the status of the task. Task forms are generated based on the task definition and deployed to the BPM Worklist Application. Based on the input data to the human task, task forms will be generated and integrated into the Oracle BPM Worklist Application.

## **Stages**

A stage is a grouping technique by which the approval process for the participant types can be organized.

## **Access rules**

Access rules defines the access levels for the actions that different stakeholders can perform on the task. This can be defined at a fine-grain as well as on a coarse-grain level in the human task definition.

## **Standalone human tasks**

Oracle SOA Suite supports creating standalone human tasks that are not linked to the BPEL process.

## **Callbacks**

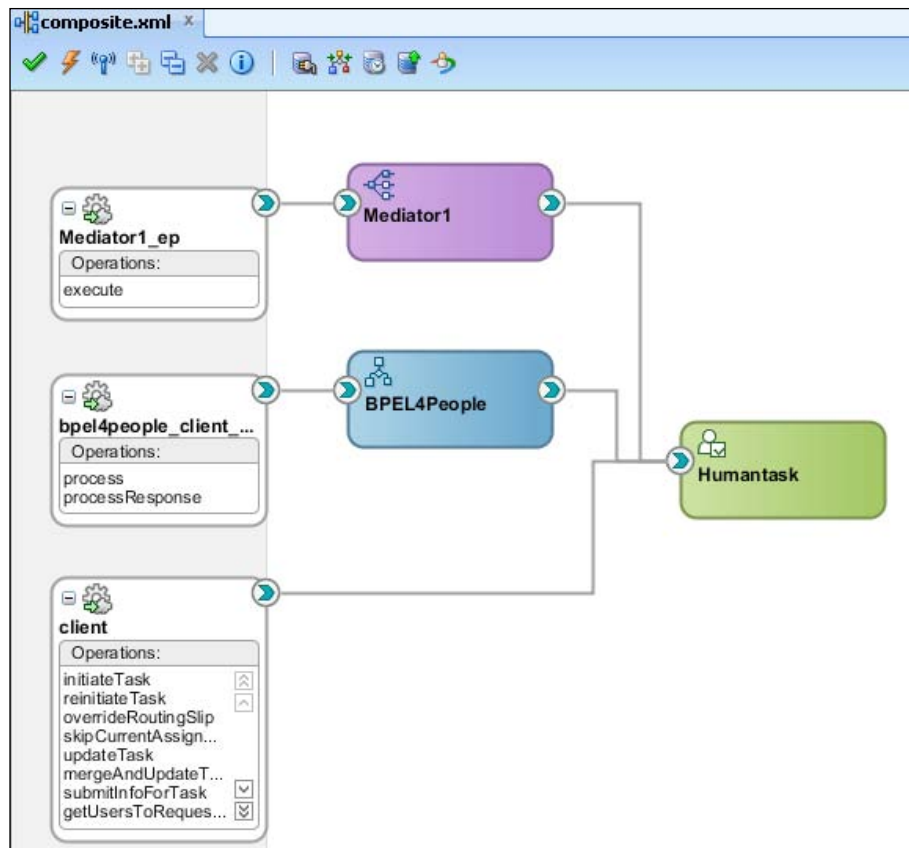
It is one of the advanced features supported by Oracle Human Workflow to include the extensibility to callback a BPEL or custom Java class to perform some advanced actions that are available within the Oracle Human Workflow framework.

## **Events**

Human Workflow has the required features to publish status events. For example, task updates, completion, and reassignment can be published to the EDN layer to facilitate the event publication and subscription.

## **Designing human tasks and services**

Human Workflow can be exposed as a service and can be integrated with Mediator or BPEL to initiate and work upon a human task, as shown in the following screenshot:



Human Workflow definition is contained in a file called `<HumanTaskComponentName>.task`.

Human Workflow definition contains eight sections to define the workflow definition. They are listed as follows:

- General
- Data
- Assignment
- Presentation
- Deadlines
- Notification
- Access
- Event

## General

The **General** section of the task definition provides features to define the **Task Title**, **Description** for the task, **Outcomes** that are expected from this task, **Priority** of the task on the overall pool of tasks, **Category** of the task, and **Owner** of the task. The following screenshot shows the **General** section of the task definition:

The screenshot shows the 'General' section of a task definition form. The left sidebar contains icons for General, Data, Assignment, Presentation, Deadlines, Notification, Access, and Events. The main area contains the following fields:

- Task Title:** A text field with a dropdown menu set to 'Text and XPath'.
- Description:** A large text area.
- Outcomes:** A text field containing 'APPROVE, REJECT'.
- Priority:** A dropdown menu set to '3 ( Normal )'.
- Category:** A dropdown menu set to 'By expression'.
- Owner:** A dropdown menu set to 'User' and a 'Static' checkbox.
- Application Context:** A text field.

## Data

The **Data** section allows the user to define the input data to the human task.

The screenshot shows the 'Data' section of a task definition form. The left sidebar contains icons for General, Data, Assignment, Presentation, Deadlines, Notification, Access, and Events. The main area contains the following fields:

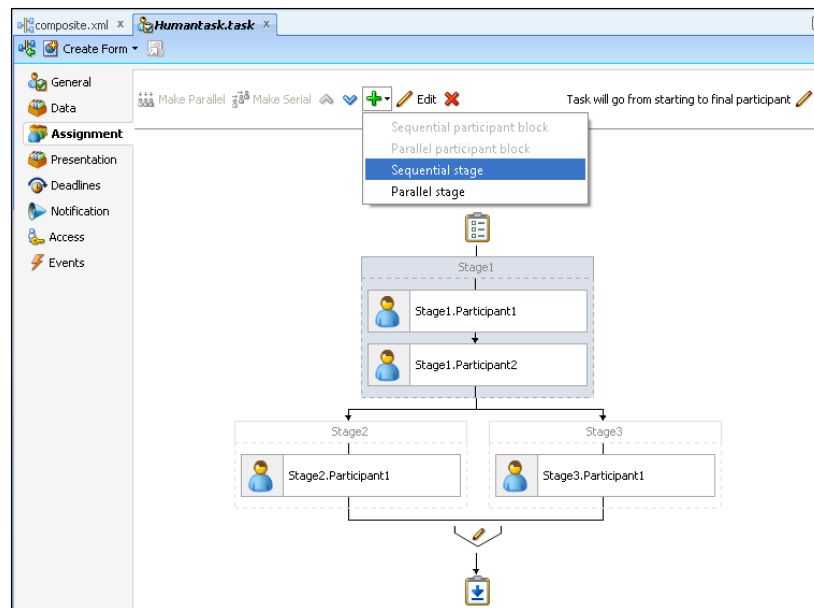
- Name:** A text field.
- ReferenceID:** A text field.
- Element or Type:** A dropdown menu.
- Mapped Attributes:** A table with columns 'Label', 'Value', and 'Description'.

Label	Value	Description
Customer Status	http://xmlns.oracle.com/pcbpel/taskservice/...	Uses customer rewards table

A context menu is open over the 'Add Task Parameter' button, showing options: 'Add string parameter', 'Add integer parameter', 'Add Task Parameter', and 'Add other parameter'.

## Assignment

The **Assignment** section of the task definition allows the developer to configure the task flow definition with different stages that can flow in parallel or sequentially. The following screenshot shows the **Assignment** section of the task definition:



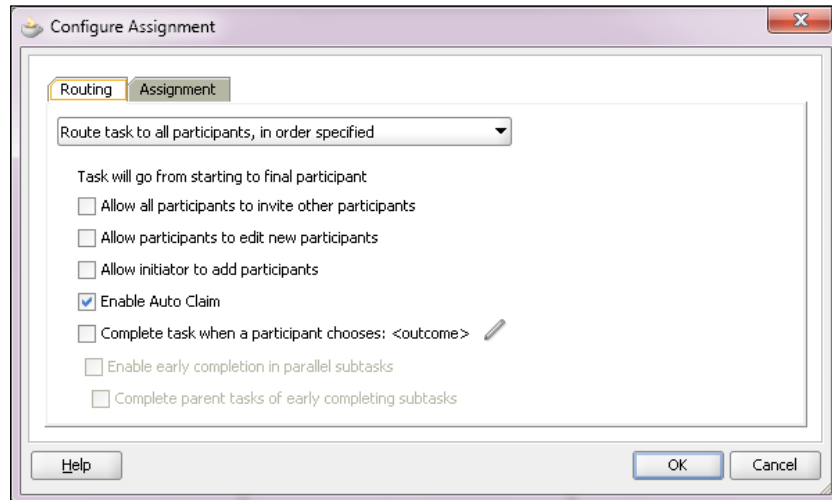
Participants can be added to the stage using the dialog box shown in the next screenshot. It provides an option to add the participant using the value of the user or rule-based assignment.

The 'Edit Participant Type' dialog box is shown. It has a 'Type' dropdown menu with options: Single (selected), Parallel, Serial, and FYI. The 'Label' field contains 'Stage1.Participant1' with a hint 'e.g., Approval Manager'. Below the dropdown is a section for 'Specify attributes using:' with radio buttons for 'Value-based' (selected) and 'Rule-based'. A table titled 'Participant Names' is visible, with columns 'Identification Type', 'Data Type', and 'Value'. The table contains one row: 'User', 'By Name', and 'Jcopper'. At the bottom, there is an 'Advanced' section with a dropdown arrow, and 'Help', 'OK', and 'Cancel' buttons.

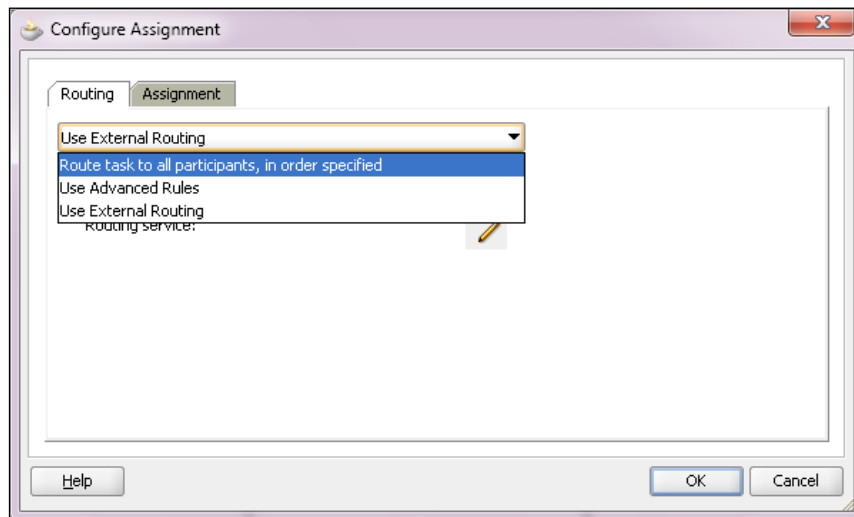
Identification Type	Data Type	Value
User	By Name	Jcopper



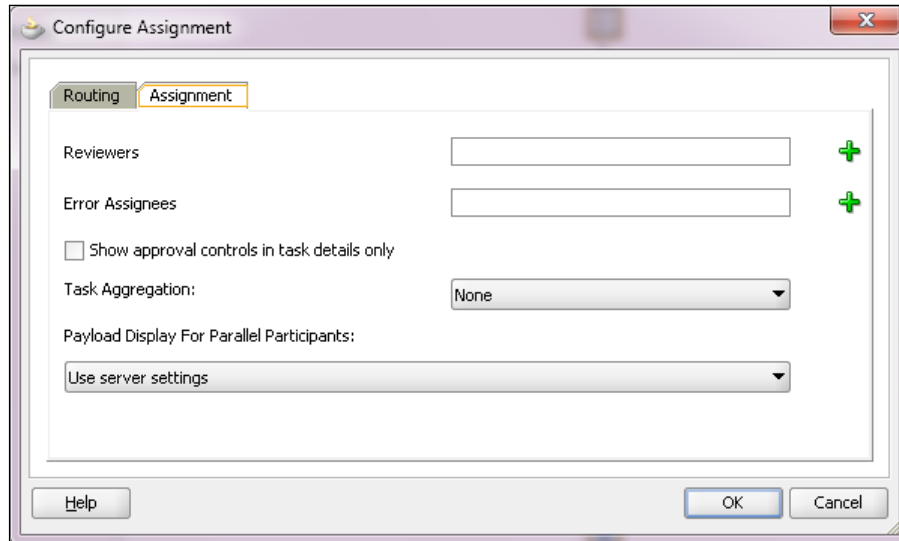
Additional configuration options to route the tasks are shown in the following screenshot. These configuration options are also available from the Composer or the Worklist Application to be changed dynamically.



**Routing** can be based on the Advanced Business Rules or using Java classes, as shown in the following screenshot:

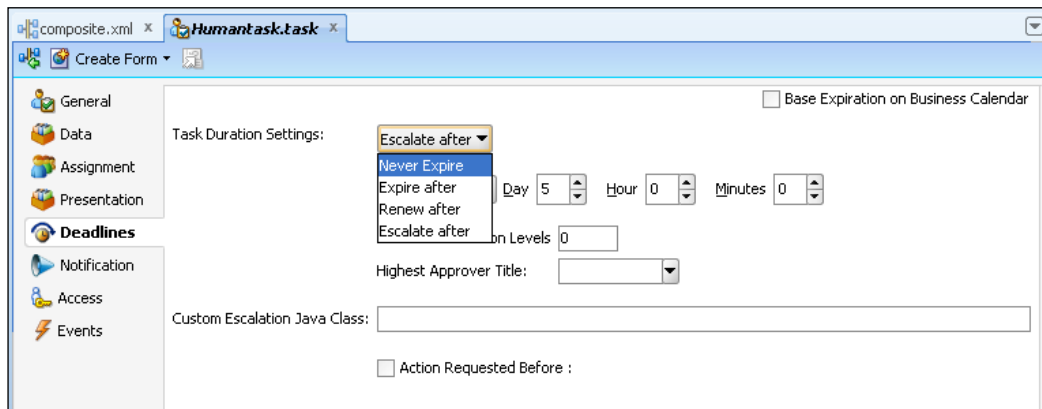


**Reviewers** and **Error Assignees** can be assigned from the **Assignment** tab, as shown in the following screenshot:



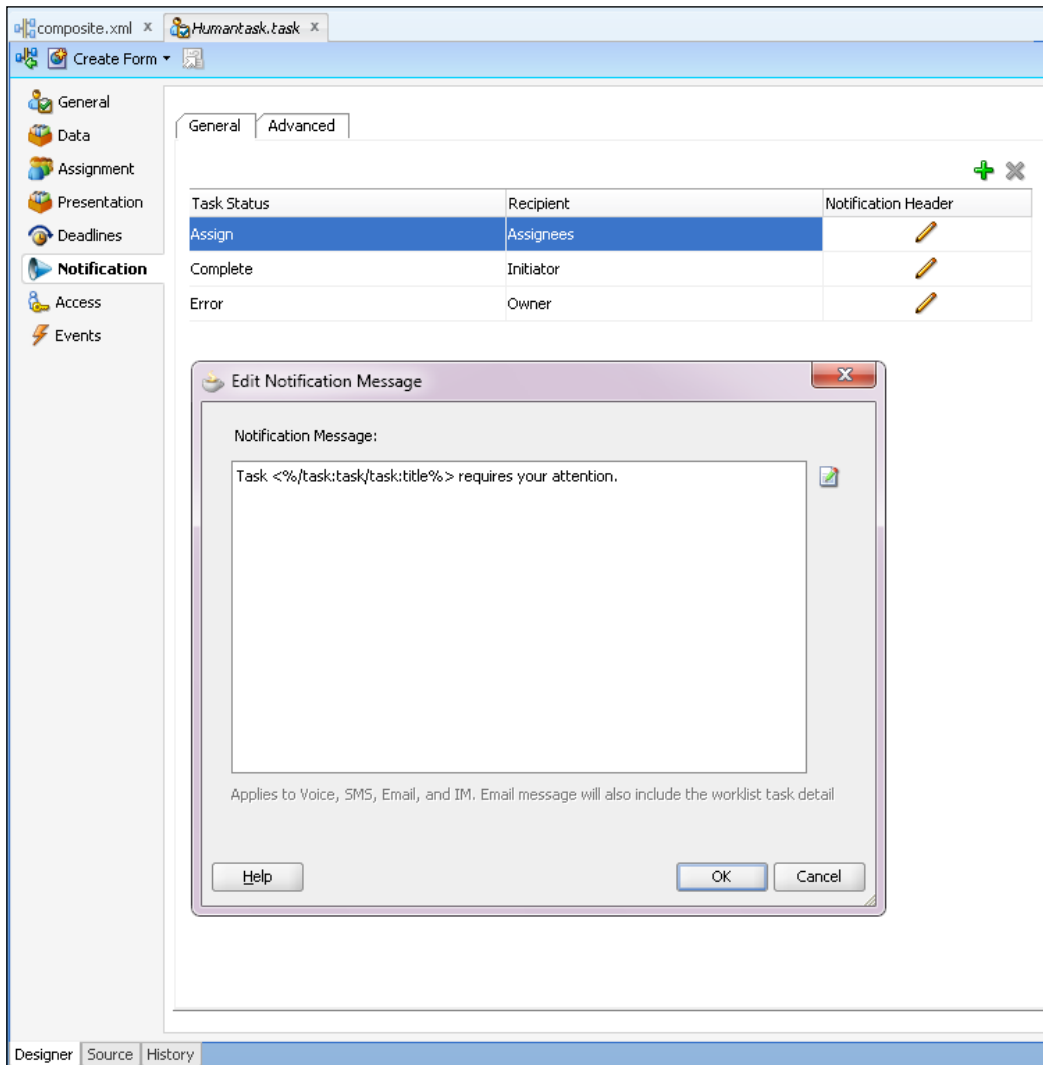
## Deadline

**Deadlines** can be specified in order to **Expire** the task, **Renew** the task, or **Escalate** the task, as shown in the following screenshot:



## Notification

The **Notification** section is used to provide assignment notification on completion or on error using predefined notification templates, as shown in the following screenshot:



## Access level

The task definition **Access** level can be defined with the help of the following two screenshots:

composite.xml x Humantask.task x

Create Form ▾

General  
Data  
Assignment  
Presentation  
Deadlines  
Notification  
**Access**  
Events

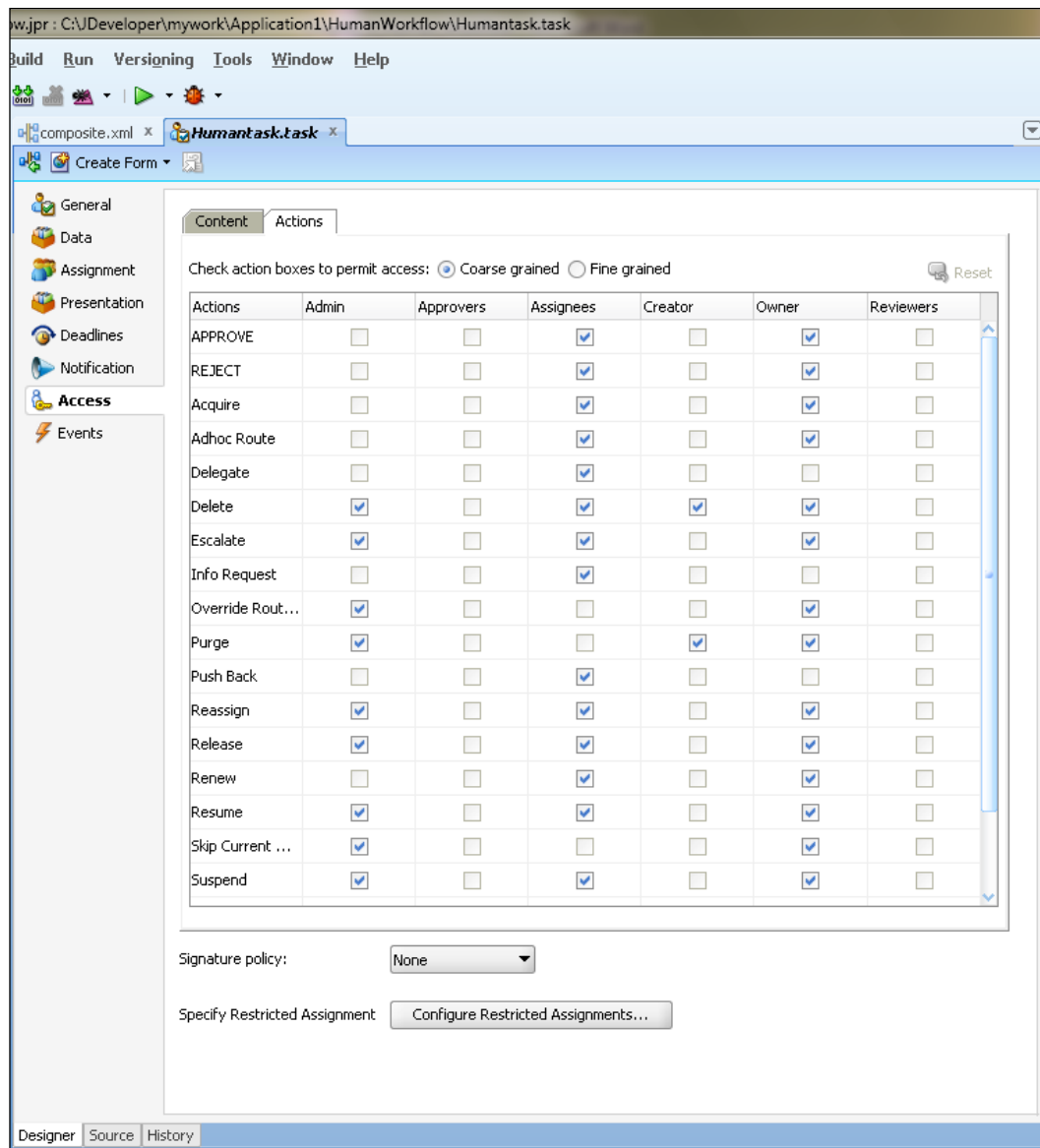
Content Actions

Set access levels for each content item: ☒ Coarse grained ☐ Fine grained Reset

Content	Admin	Approvers	Assignees	Creator	Owner	Reviewers
Assignees	Read	Read	Read	Read	Read	Read
Attachments	Read	Read	Write	Write	Write	Write
Comments	Read	Read	Write	Write	Write	Write
Dates	Read	Read	Read	Read	Read	Read
Flexfields	Read	Read	Write	Write	Write	Read
History	Read	Read	Read	Read	Read	Read
Payload	Read	Read	Write	Write	Write	Read
Reviewers	Read	Read	Read	Read	Read	Read

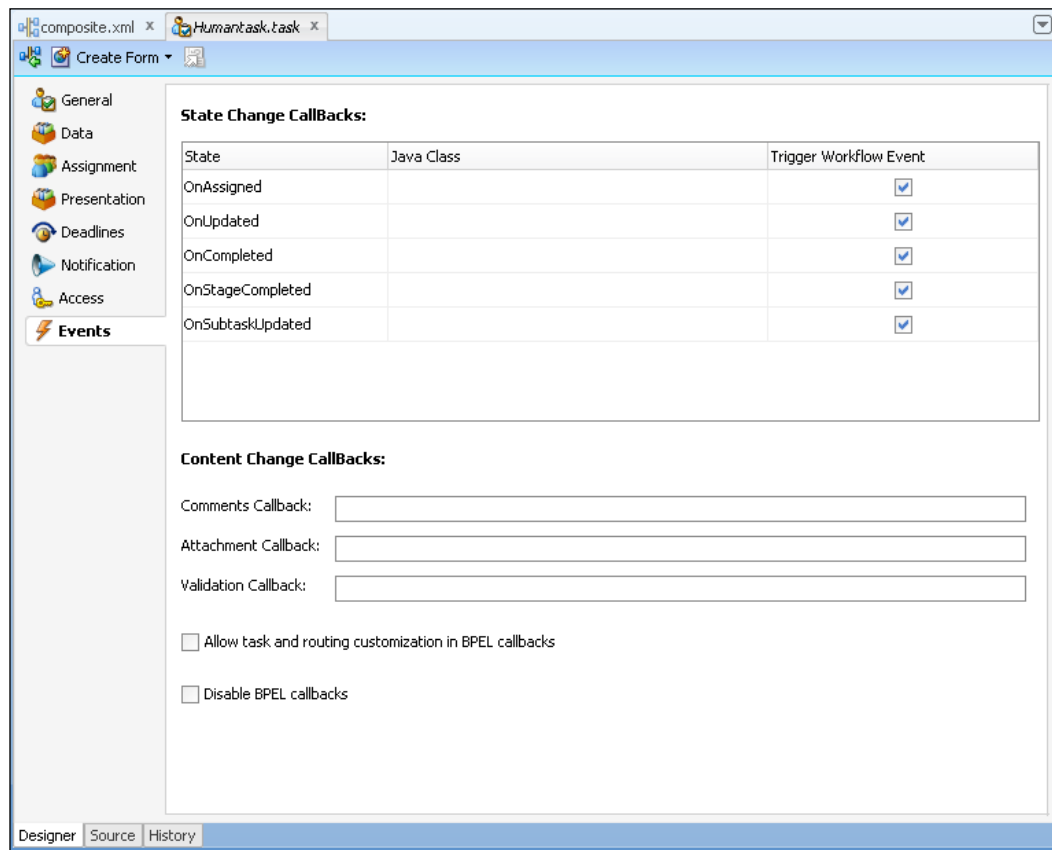
Signature policy: None ▾

Specify Restricted Assignment Configure Restricted Assignments...



## Events

**Events** can be generated on assignment, on update, on completion, on stage completion, and on sub task update, as shown in the next screenshot. These events can be subscribed through the EDN layer to take the required action.



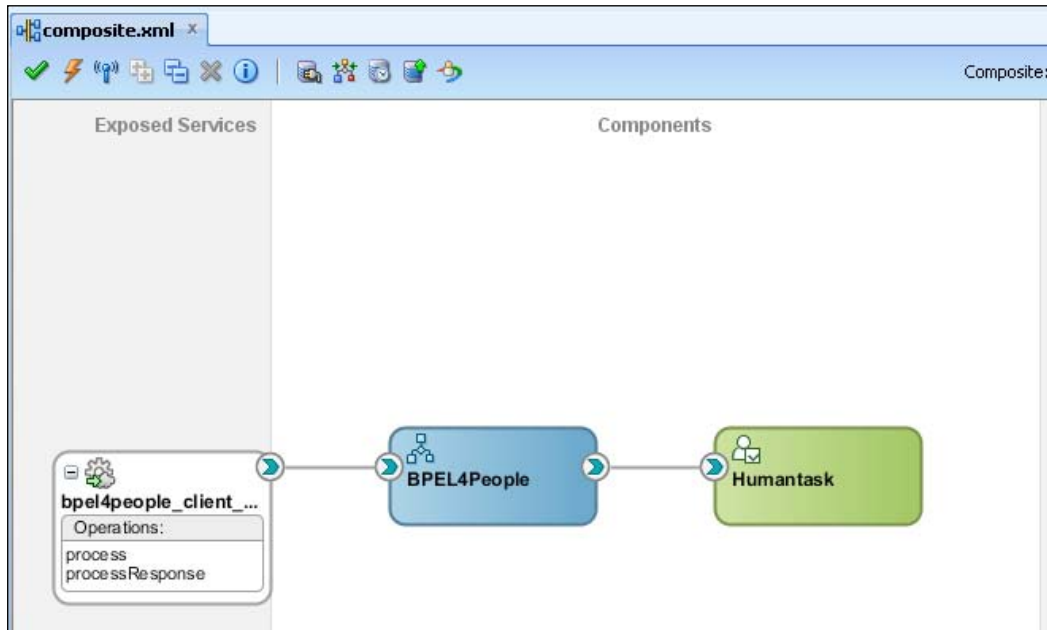
## Invoking a human task from a BPEL process

Invoking a human task from a BPEL process is a two-step process:

1. Creating a human task.
2. Wiring and invoking the human task from BPEL.

The following are the steps to create a sample human task:

1. Open `composite.xml` of the composite.
2. Drag-and-drop a human task component onto the composite.
3. Specify the human task name, change the namespace value, and click **OK**.
4. Wire the BPEL process to the human task:



5. Double-click the human task to open the task definition editor:

Task Title: Text and XPath | ApproveHumanTask

Description:

Outcomes: APPROVE,REJECT

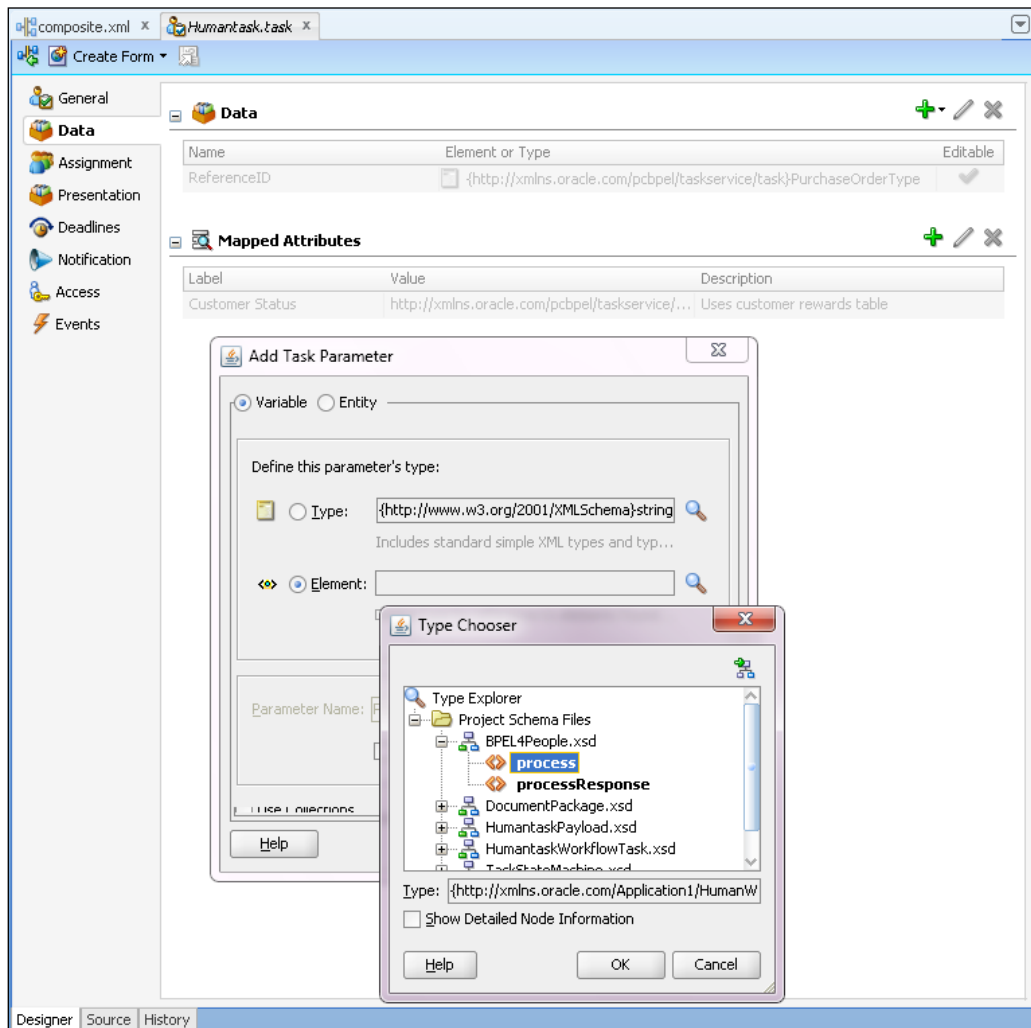
Priority: 3 (Normal)

Category: By expression

Owner: User

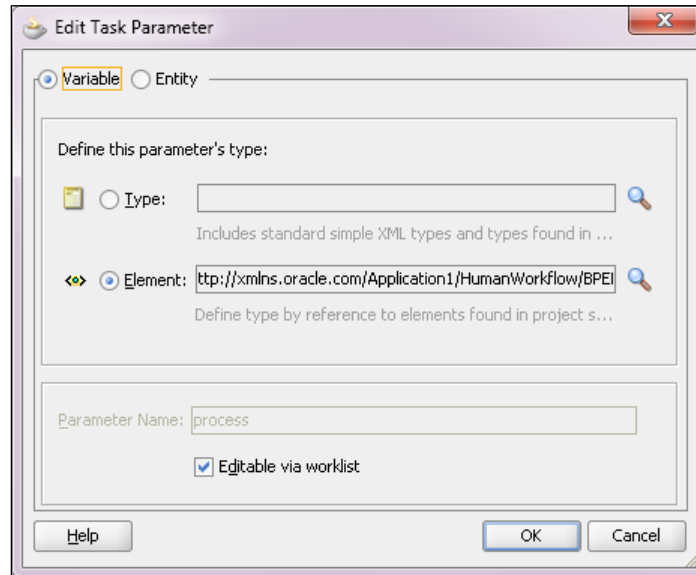
Application Context:

6. Specify the following settings:
  - Title drop-down as **Text and Xpath**
  - **Task Title:** Enter ApproveHumanTask and click **OK**
  - **Description:** Manual approval task for large orders
7. Specify the task data to be used by the human task. Choose the element **Type** and choose the **Element** that would contain the data to be used by the human task:

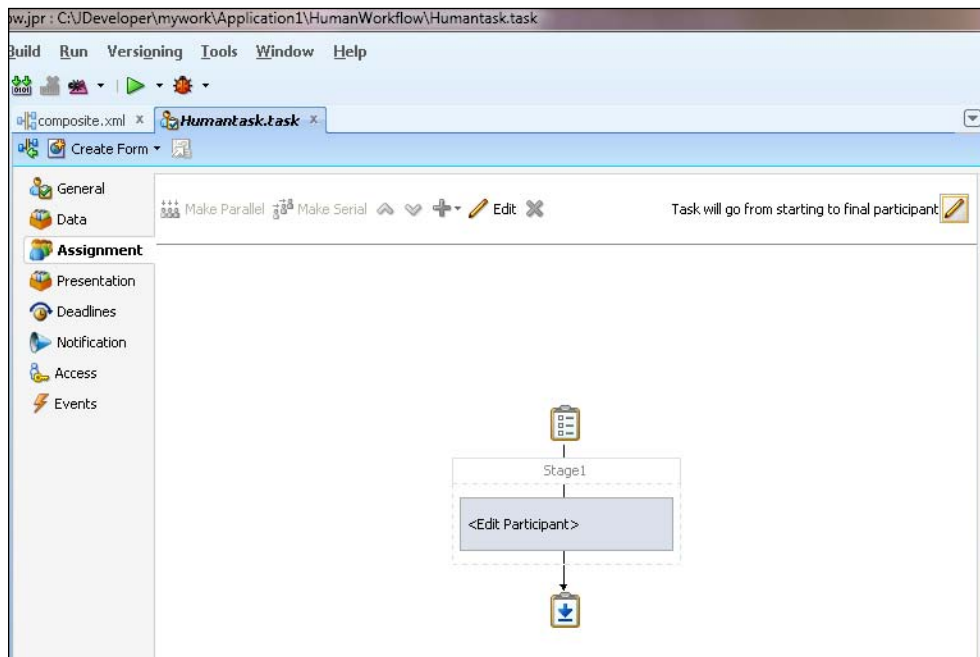




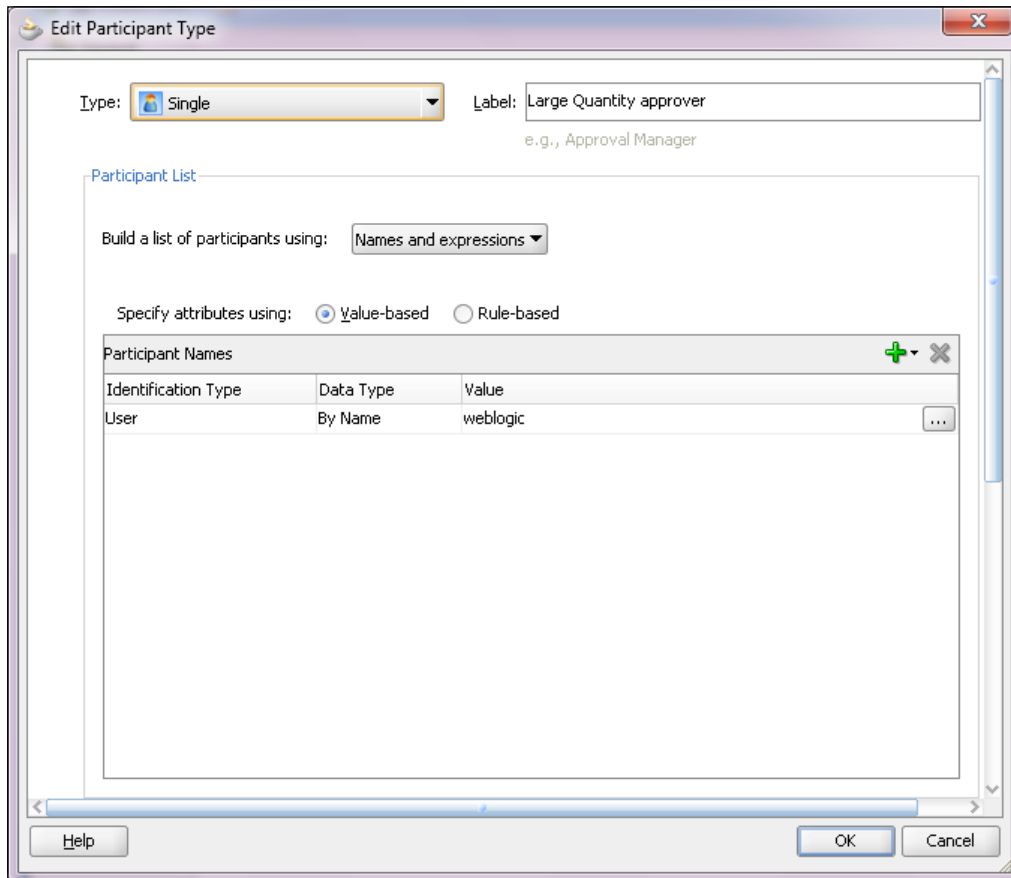
8. Keep the **Editable via worklist** checkbox as checked and click **OK**:



9. In the **Assignment** section, double-click on the **<EditParticipant>** box, as shown in the following screenshot:



10. In the **Edit Participant Type** dialog, specify the following:
  - **Type:** **Single** and **Label:** Large Quantity approver



11. Use the green + sign next to the **Participant Names** to add a user.
12. Set the **Identification Type** to **User** (already set).
13. Set the **Data Type** to **By Name** (already set).
14. In the **Value** field, enter `weblogic` (no quotes).
15. Save all the components, close the task definition editor, and return to the composite.

## Calling the human task from BPEL

1. Wire the human task from BPEL.
2. Double-click on the BPEL component to open the BPEL designer.
3. Drag an invoke activity, add it to BPEL, and connect to the Human Workflow partner link.
4. Assign the required input to the Human Workflow service.



The steps required to achieve this will vary with different versions of Oracle SOA Suite 11g. Please use the appropriate steps to connect the human task with BPEL.

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Describe Human Workflow architecture, features, and concepts:** We discussed in detail about the Oracle Human Workflow
- **Design human tasks and services:** We discussed in detail about configuring Oracle human task and associated services
- **Invoke a human task from a BPEL process:** We understood the steps involved in invoking the human task from a BPEL process

## Self-review questions

1. Identify different ways of assigning tasks to a user. Choose the right answers from the following list:
  - a. Static
  - b. Dynamic using XPath expression
  - c. Rule-based
  - d. None of the above

2. Identify the types of deadlines associated with a task. Choose the best answers from the following list:
  - a. Reminders
  - b. Escalations
  - c. Renewal
  - d. Expiration
  - e. a, b, and c are the right options
3. Identify the notification types associated with a task. Choose the best answers from the following list:
  - a. SMS
  - b. Voice
  - c. E-mail
  - d. All of the above
  - e. None of the above
4. Business Rules can be used to route the task to a required user, group, or application role.
  - a. True
  - b. False
5. Oracle Worklist can be used to route the task to a required user, group, or application role.
  - a. True
  - b. False
6. Human Workflow can be configured using which of the following options:
  - a. Workflow management utilities
  - b. Task routing services
  - c. Identity services
  - d. Security services
  - e. Worklist Application

7. The Oracle BPM Worklist Application can be used for \_\_\_\_\_. Choose the right answers from the following list:
  - a. Defining the Human Workflow
  - b. Configuring the Human Workflow
  - c. Task management
  - d. Defining workflow delegation rules
  - e. Personalizing the task flow
8. Oracle Worklist Application can be used for \_\_\_\_\_. Choose the right answers from the following list:
  - a. Defining the Human Workflow
  - b. Configuring the Human Workflow
  - c. Task management
  - d. Defining workflow delegation rules
  - e. Personalizing the task flow
9. Identify the default list of out-of-the-box task analysis reports from the following list:
  - a. Unattended tasks
  - b. Tasks priority
  - c. Task cycle time
  - d. Task productivity
  - e. All of the above
10. Identify the list of out-of-the-box Oracle BPM task analysis reports from the following list:
  - a. Tasks routing
  - b. Tasks priority
  - c. Task cycle time
  - d. All of the above
11. Oracle Workflow application provides features to export and import the user-related configuration from one environment to another.
  - a. True
  - b. False

12. Identify the Human Workflow services provided by Oracle Human Workflow architecture. Choose the right answers from the following list:
  - a. Task service
  - b. Identity service
  - c. Notification service
  - d. Runtime service
  - e. User metadata service
  - f. Evidence service
13. Identify the list of task statuses provided by the Oracle BPM Worklist Application. Choose the right answers from the following list:
  - a. Assigned
  - b. Suspended
  - c. Completed
  - d. Expired
  - e. Stale
  - f. All of the above
14. Oracle Business Rules can be used to alter the routing flow in the Human Workflow.
  - a. True
  - b. False
15. Human task can be exposed as a service.
  - a. True
  - b. False
16. Oracle Mediator can connect to Oracle Human Workflow. Choose the correct answers from the following list:
  - a. True
  - b. False

17. Identify the default outcomes of Oracle Human Workflow.

- a. Approve, Reject
- b. Accept, Reject
- c. Yes, No
- d. Okay, No
- e. True, False
- f. None of the above

# 9

## Business Rules

Rules and policies are part of all the business processes defined in different zones of business. Defining Business Rules and Policies helps in the effective execution of business and it creates a reliable business environment. Business Rules are defined by the organization as well as by the government to conduct business in an ethical and moral way. Business Rules are also created to encourage businesses to be more competitive as well as to identify the right customers and serve them with more care. The following points are examples of business rules defined in different zones of the business environment:

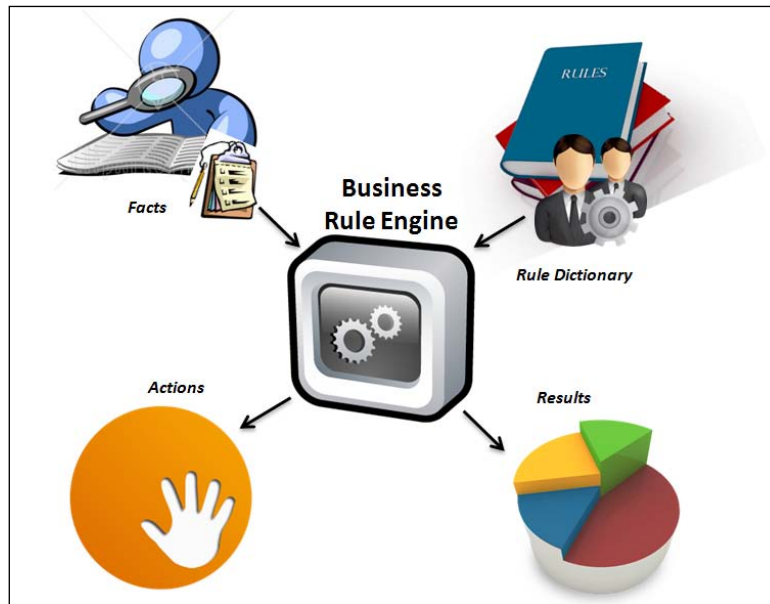
- If the total annual income of a family is greater than \$80,000, then auto-approve a loan less than \$120,000 with 3 percent interest
- If the age of a customer is less than 18, then decline to sell products that are classified under Section 18 of Govt. Act 1937
- If a traveler account has total miles for the year greater than 500,000, then change the customer status to Platinum
- If a customer's payment history is excellent and rated as Gold, provide 5 percent additional discount on the new services subscribed by the customer for the next nine months

Demand for transparency in business from customers, employees, partners, governments, and external auditing firms are increasing every day due to globalization of organizations. This demand calls for centralizing the business rules to create healthy and valuable competition in the conduct of business. Standardized Business Rules help the customer to understand the business practises across different zones of life and choose the right products and services that suits their needs.



Competition in the business environment demands the organization to have an agile business process definition; it opens the doors for innovative thinking and calls for swift implementation of new business process definitions through the right agile business technology. **Oracle Business Rules** provide these features and they will be discussed in detail in this chapter of the book.

The Oracle Business Rules Framework provides the following required features to enable agility and transparency in business conduct:



- Ability to define the centralized Business Rules that can be accessed by a wide range of technologies and modified by business users in a non-technical language.
- Ability to change the Business Rules on-demand without going through the pains of software life-cycle management procedures to enable a business change request.
- Ability to integrate the Business Rules with business processes to enable agility in business process execution.
- Ability to access the centralized Business Rule definition from Java and SOA components to provide extended integration capabilities from other layers of business applications.
- The Oracle Rules Engine adopts RETE algorithm to enable hyper parallelism and scalability. A wide range of rules can be defined through if-then-else and Decision Tables.

This chapter will cover the following list of exam objectives:

- Describe the Oracle Business Rules Architecture and the Oracle Rule Engine details
- Explain the Business Rules concepts
- Create rules with the JDeveloper Rules Designer
- Test Business Rules from JDeveloper
- Integrate a simple rule with a BPEL process
- Edit business rules from the SOA Composer

## The Oracle Business Rules Architecture and Oracle Rule Engine details

The **Oracle Business Rule Architecture** consists of the components listed in the following image. Each of these components is used for a different purpose in different layers of the Oracle Business Rule Architecture. Rule Author and SDK are used as design-time components. Rule Language and Rule Engine are used as runtime components that interact with decision components developed from design-time components:

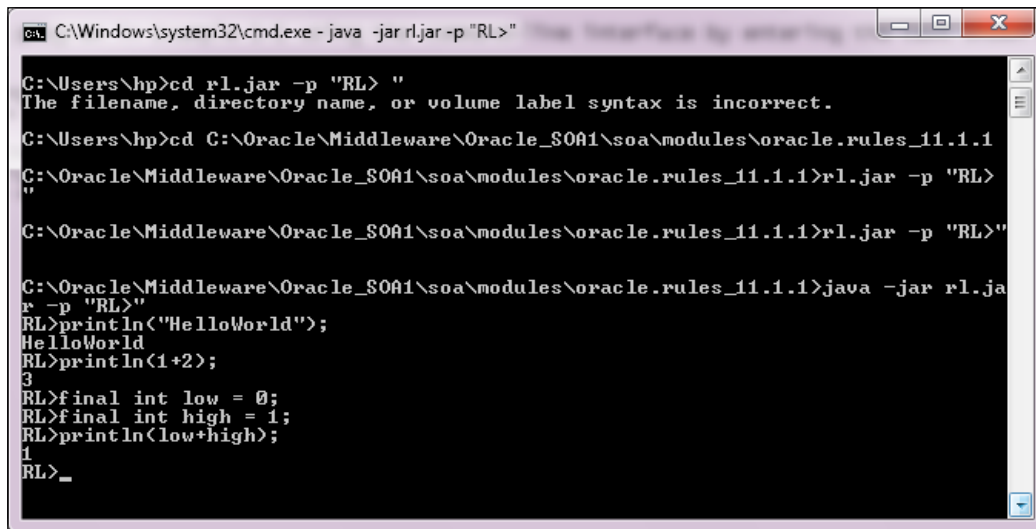


The descriptions of the components are given as follows:

- **Rule Author:** Rule Author or SOA Composer for Business Rules is shipped along with Oracle SOA Suite to define the Business Rule without knowing any programming language. This helps the business analysts to define the Business Rules without the help of software developers, implement the Business Rule changes on-demand, and expect the immediate results and changes implemented. The Oracle BPM product provides the features to forecast the changes and stimulate a process with change in rule impact using Process Stimulation workspaces.

- **Rule SDK:** Rule SDK is a Java library through which developers can write custom rule programs. Rule Author makes use of Rule SDK to create, modify, view, or delete the Business Rules.
- **Rule Language:** RL is a Java-like language with rich syntax for Oracle Business Rule programs. Using RL, application programs can assert Java objects as facts and rules can reference these properties in the rule engine to execute the rules defined in the rule dictionary.
- The following code snippet shows the command-line utilities that can be used to test the Rule Engine:

```
C:\Users\hp>cd C:\Oracle\Middleware\Oracle_SOA1\soa\modules\
oracle.rules_11.1.1
C:\Oracle\Middleware\Oracle_SOA1\soa\modules\oracle.rules_11.1.1>
java -jar rl.jar -p "RL>"
RL>println("HelloWorld");
HelloWorld
RL>println(1+2);
3
RL>final int low = 0;
RL>final int high = 1;
RL>println(low+high);
1
RL>
```



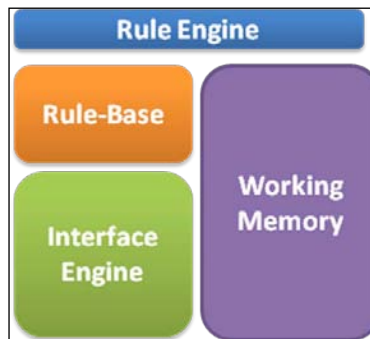
The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe - java -jar rl.jar -p "RL>"". The window contains the following text:

```
C:\Users\hp>cd rl.jar -p "RL> "
The filename, directory name, or volume label syntax is incorrect.

C:\Users\hp>cd C:\Oracle\Middleware\Oracle_SOA1\soa\modules\oracle.rules_11.1.1
C:\Oracle\Middleware\Oracle_SOA1\soa\modules\oracle.rules_11.1.1>rl.jar -p "RL>
"
C:\Oracle\Middleware\Oracle_SOA1\soa\modules\oracle.rules_11.1.1>rl.jar -p "RL>"

C:\Oracle\Middleware\Oracle_SOA1\soa\modules\oracle.rules_11.1.1>java -jar rl.jar
-p "RL>"
RL>println("HelloWorld");
HelloWorld
RL>println(1+2);
3
RL>final int low = 0;
RL>final int high = 1;
RL>println(low+high);
1
RL>_
```

- **Rule Engine:** Oracle Rule Engine is a Java library that effectively applies the rules to facts and defines the process rules. The Rule Engine makes use of the RETE algorithm to effectively execute the rules and matches it with facts. Rule Engine also supports developers' interactions through command-line utilities:



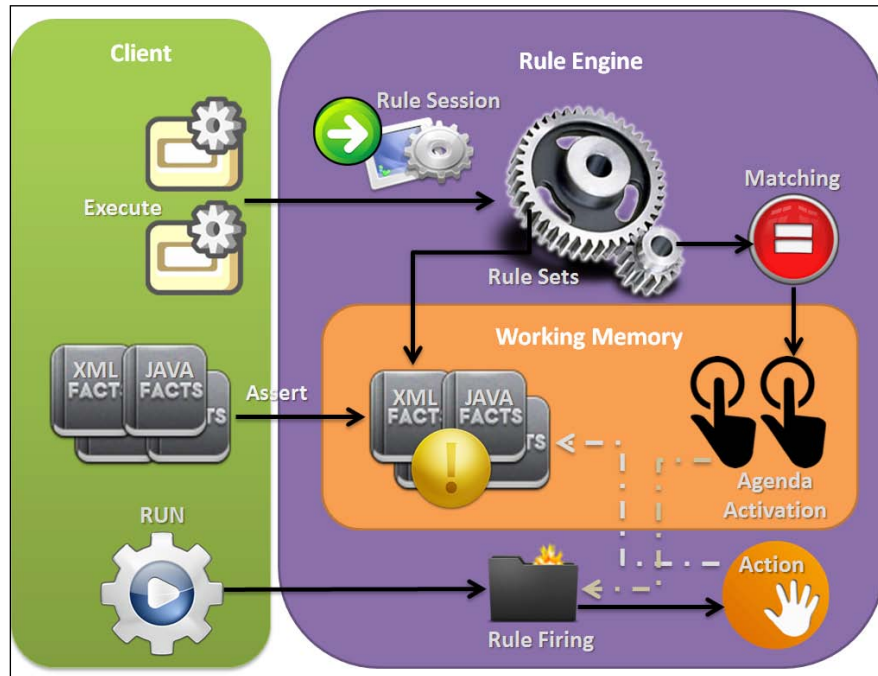
- **Rule-Base:** This contains the rules to be executed against the facts that are asserted in the engine.
- **Interface Engine:** It processes the rules and matches the respective facts for a given run. It utilizes the facts in the working memory.
- **Working Memory:** Stores the facts that are asserted in the Rule Engine through assert calls.

## The Rule Engine working model

When facts are modified in the working memory of the business Rule Engine:

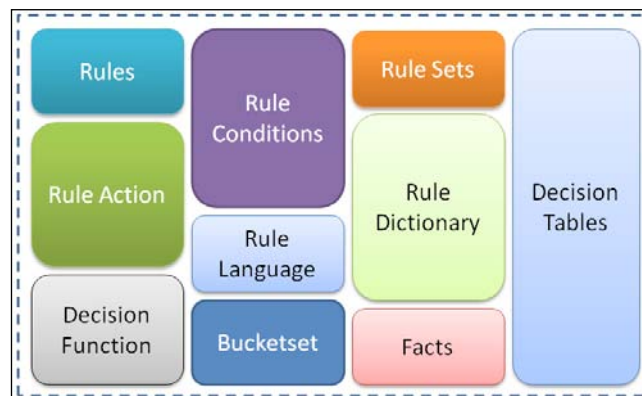
1. Conditions for rules are evaluated and matched.
2. Matching rules are added to the agenda and pooled for activation.
3. Rules that are no longer matched are removed from the agenda step.
4. The Rules Engine runs and executes actions for activated rules that matched the asserted facts.

The Rule Engine working model is shown in the following image:



## Oracle Business Rules concepts

This section will highlight the key concepts and terms to be understood about the Oracle Business Rules:

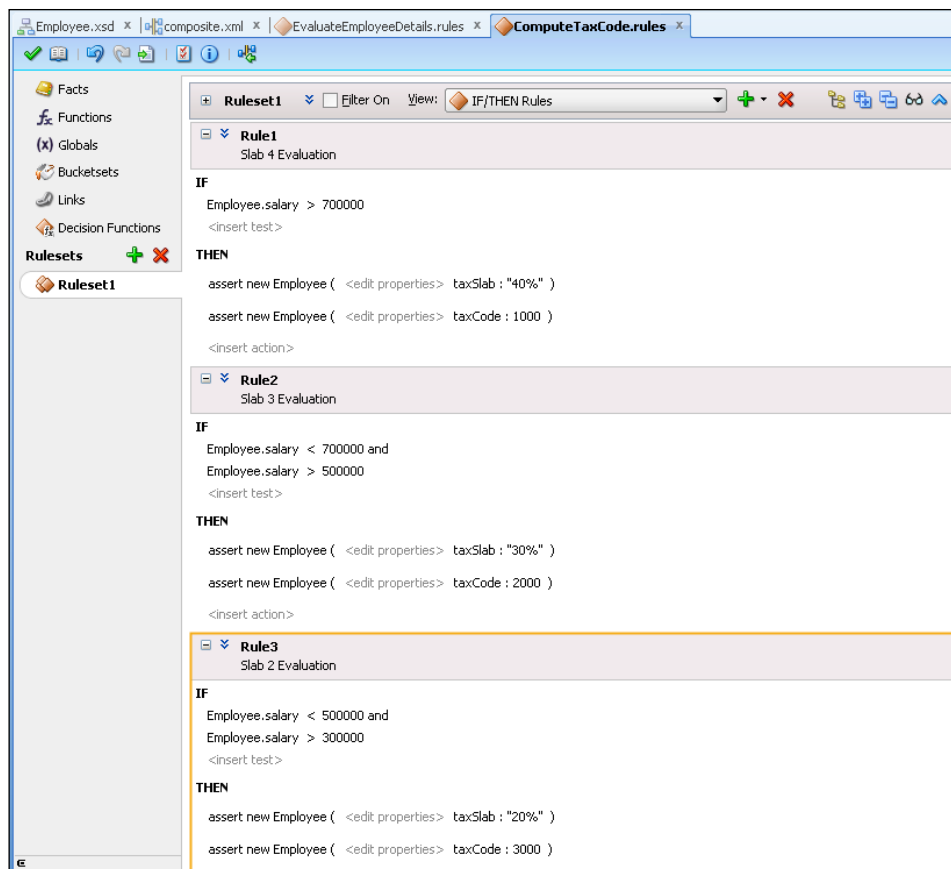


- **Rules:** Rules are discrete statements or configurations in the Rule Dictionary that reflect the Business Rules or Business Policy. Rules can be used to take action or validate the data. It is recommended that the Rules are predominately used for evaluating the facts to determine the actions to be performed. Rules describe the policy with the required action to be taken for the facts that meet the conditions specified.

Example:

```
IF
Employee.salary < 70000
Then
assert new Employee (taxSlab : "40%")
assert new Employee (taxCode : 1000)
```

- The following screenshot shows the tax computation done by Business Rules by evaluating the facts and applying the underlying Business Rules for tax computation:



- **Rule Condition:** Rule Condition is the statement that describes the evaluation criteria to meet the action.

Example:

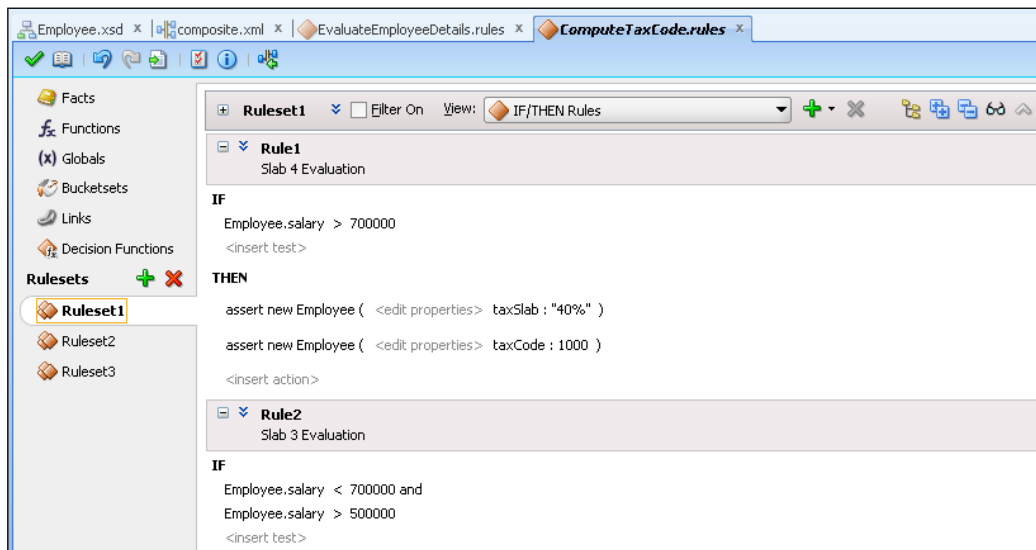
```
IF  
Employee.salary<70000 and Employee.salary> 50000
```

- **Rule Action:** Rule Action is the statement that describes the action to be performed when the Rule Condition is met.

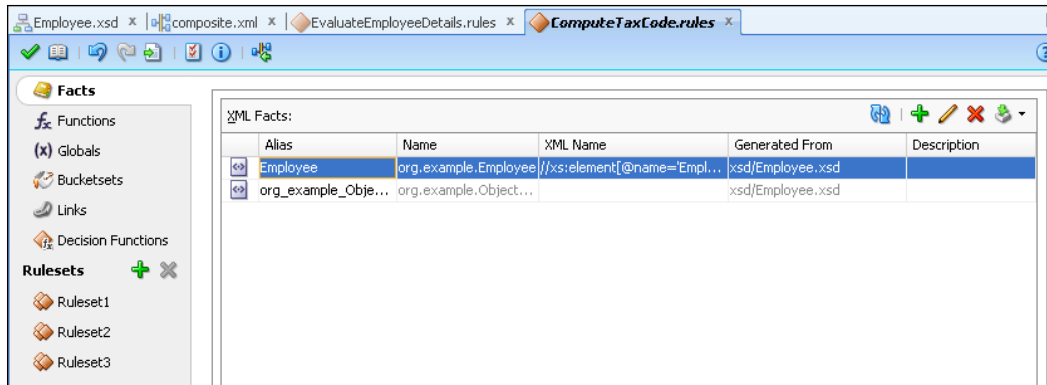
Example:

```
assert new Employee (taxSlab : "40%")  
assert new Employee (taxCode : 1000)
```

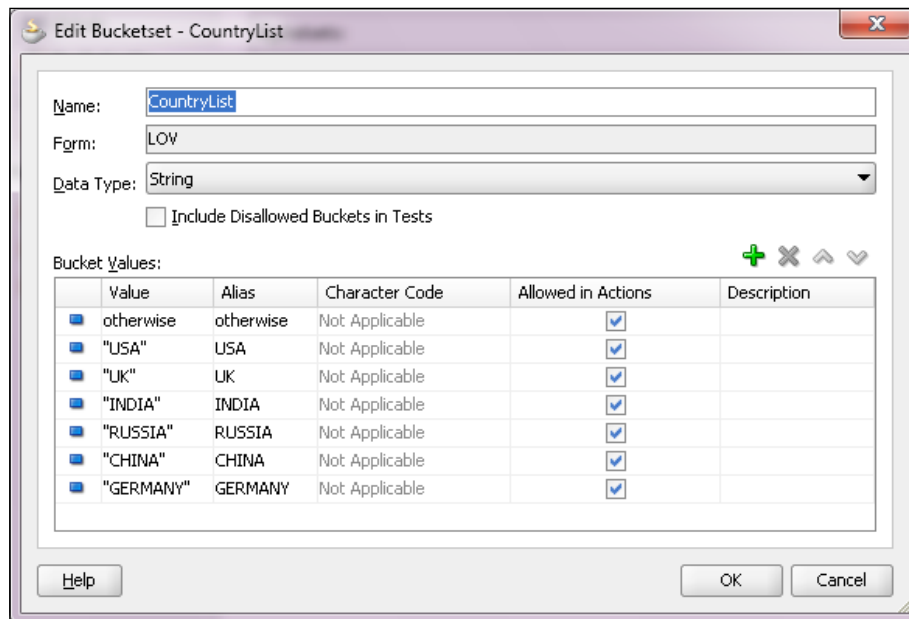
- **Rule Set:** Rule Set is the collection of rules and decision tables that can be accessed externally through Java or SOA components. Related groups can be grouped using Rule Set and Order for Rule Firing can be defined using the Rule Set definition:



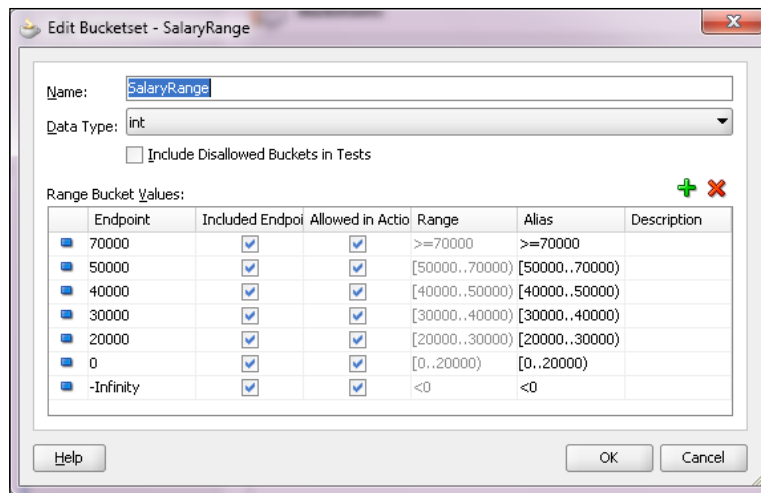
- **Facts:** Facts are objects on which the rules will be evaluated to produce required results on which an action can be performed. The following screenshot shows the XML facts on which rules can be executed. Oracle Business Rules support XML, Java facts, RL facts, and ADF-BC facts.



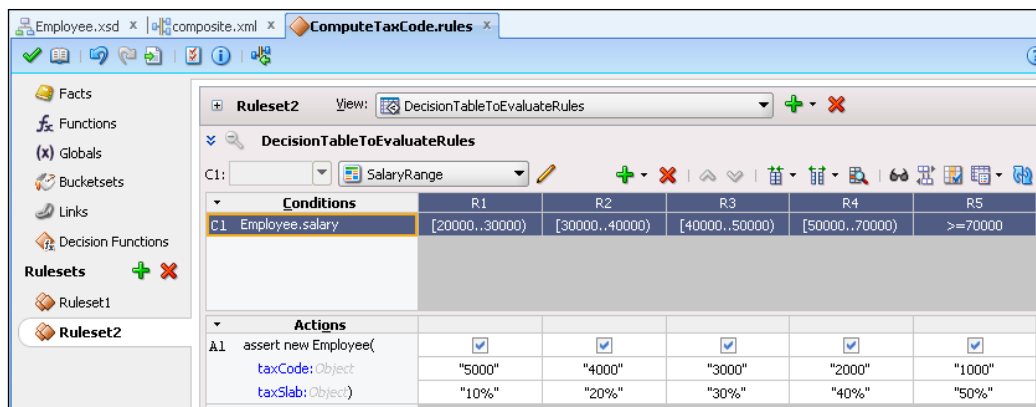
- **Bucketset:** List of Values and List of Ranges to be used in evaluating the rules are placed into a Rule Dictionary using Bucketset. These values can be used in the action for evaluation, asserting or setting values to the field. Examples for Bucketset are shown in the following screenshots:





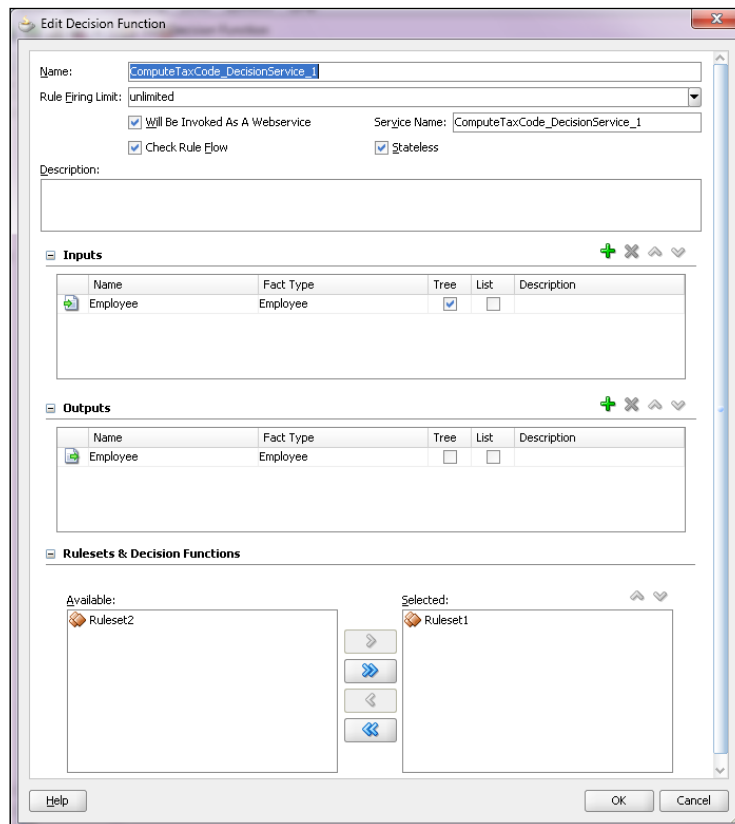
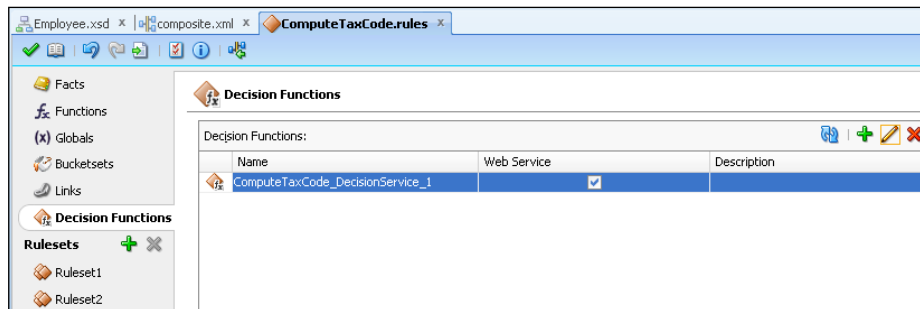


- **Decision Tables:** A Decision Table is a form of representing Business Rules. The following screenshot shows the Business Rules' implementation using Decision Tables:



- **Rule Dictionary:** Rule Dictionary is a collection of facts, functions, Bucketsets, Rule Sets, and Decision Functions. A Rule Dictionary is an XML file that stores the previously-specified information with the \*.rules extension. Rule files can be located under <BusinessRulesProject>\oracle\rules\businessrules with the \*.rules file.
- **Rule Language:** Rule Language provides features to define the valid syntax for Oracle Business Rules programs. Programmers can make use of RL as a full-featured programming language and business users can make use of Rule Author to author the Business Rules and need not interact with RL.

- Decision Function:** Decision Function is the technique by which the rules can be exposed to the external world. It can be accessed by SOA or Java components. The order of the RuleSet to be executed can be determined using Decision Functions. Multiple Decision Functions can be exposed from Business Rules and it can be grouped accordingly, based on the complexity of the Business Rule to be implemented. There are two types of Decision Functions used in Oracle Business Rules; they are stateful and stateless:



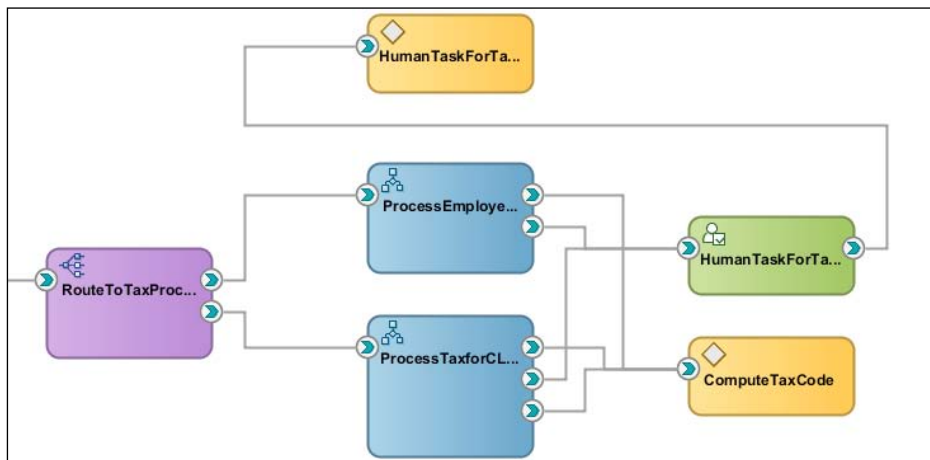
## Creating rules in JDeveloper Rule Designer

Creating rules in JDeveloper is a new feature introduced in JDeveloper 11g and it is deployed in Oracle SOA Suite as a composite to be executed by the SOA-Infra through the Business Rule Engine. Oracle Business Rules can be a standalone component or it can be integrated with Mediator, BPEL, or human task for **Dynamic Content Based Decision Making**.

The following images show all the variations in which Business Rules can be integrated with different Oracle SCA components.

- BPEL using Business Rules to evaluate Business Rules
- Human task using Business Rules for Dynamic Assignment of Participants

These are shown in the following image:



- Mediator using Business Rules to achieve **Dynamic Routing** through Business Rules is shown in the following image:



The following list of items should be determined before creating Business Rules:

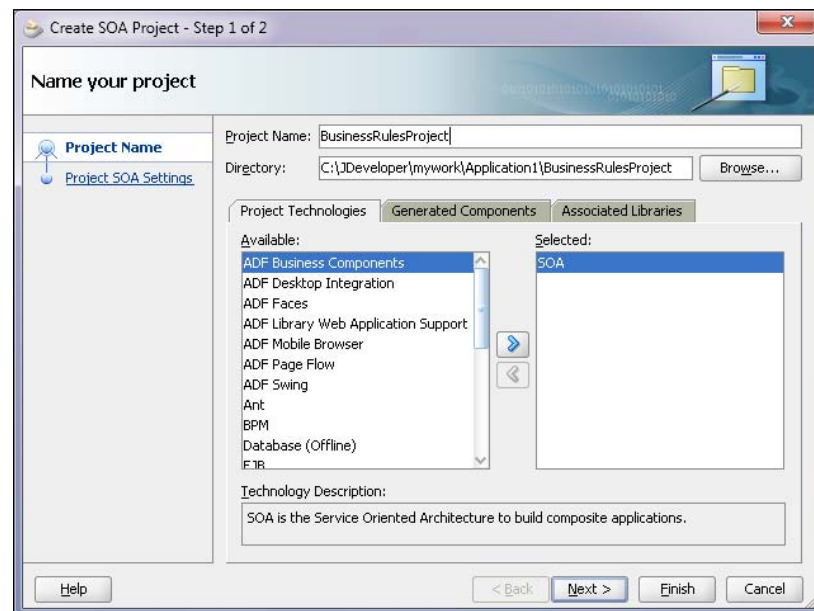
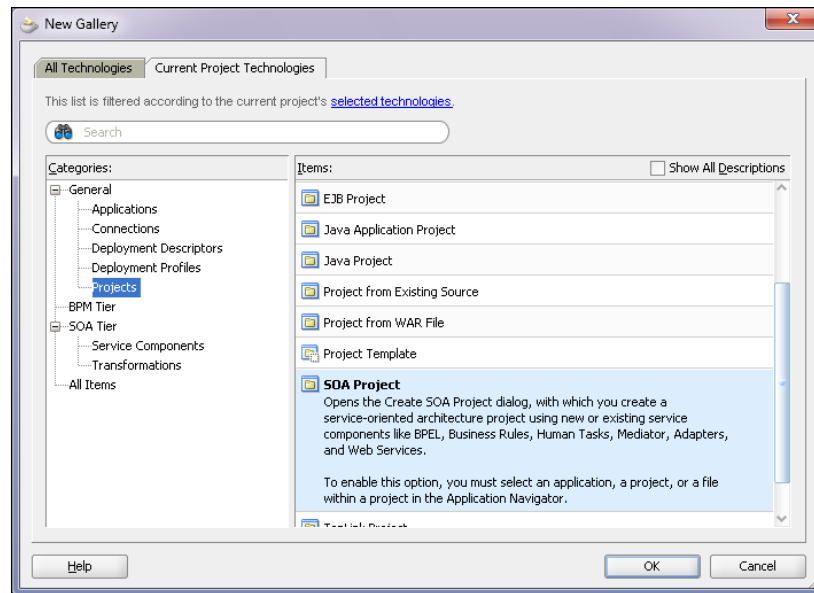
1. Define the input message to be used for the Business Rules.
2. Identify the **Facts** to be provided for the input message.
3. Identify the Business Rules to implement.
4. Identify the **List of Values** or **List of Ranges** to be used in Business Rules.
5. Identify the RuleSets to be defined that would group the rules.
6. Identify the functions to be defined to be used as a part of the Business Rules.
7. Identify the dictionaries to be linked to the Business Rules, if any.
8. Define the state of a Business Rule and expose it as a service to access the Business Rule through the SOA-infra layer in Oracle SOA Suite.
9. Oracle Business Rules can be connected to Oracle SCA components through wires in `composite.xml` and achieve the required dynamic behavior.

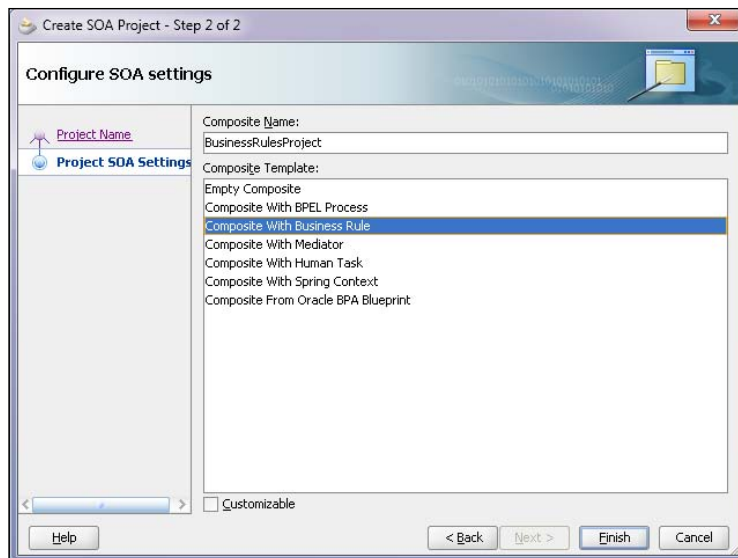
Creating Business Rules or adding a Business Rule can be done in different ways in JDeveloper based on the requirement:

1. A new Oracle SOA project with a Composite with Business Rule Template can be used to create the New Oracle SOA Composite with Business Rules.
2. Drag-and-drop the Business Rule component into `composite.xml`.

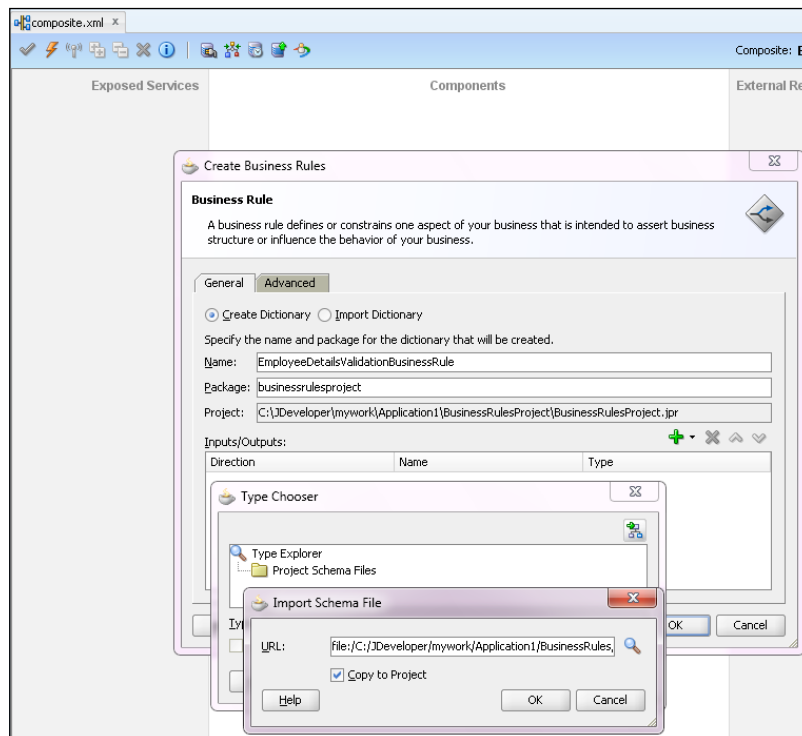
The following example shows a simple Business Rule created from JDeveloper with if-then using the Decision Table style:

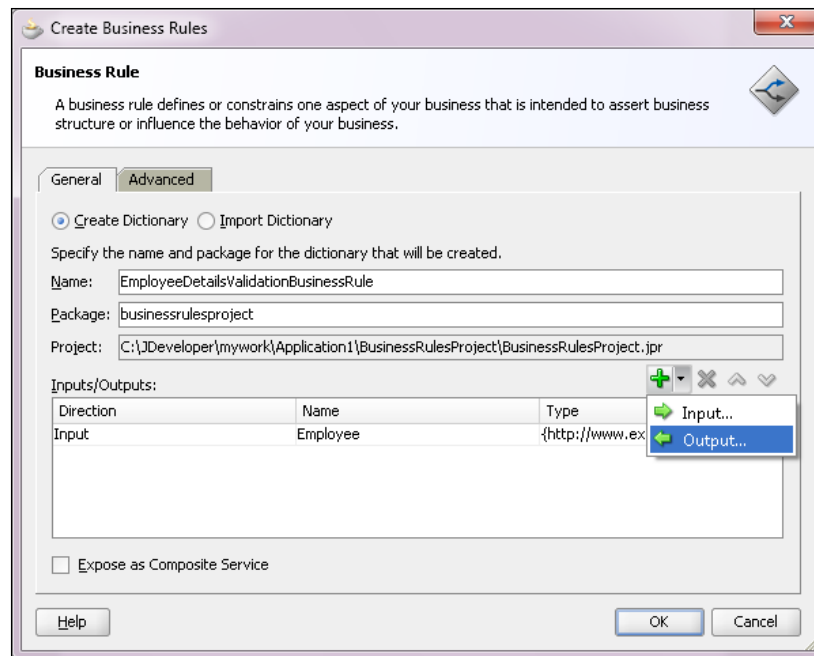
1. Create a new SOA project using Composite with a Business Rule as the Template, as shown in the following screenshots:



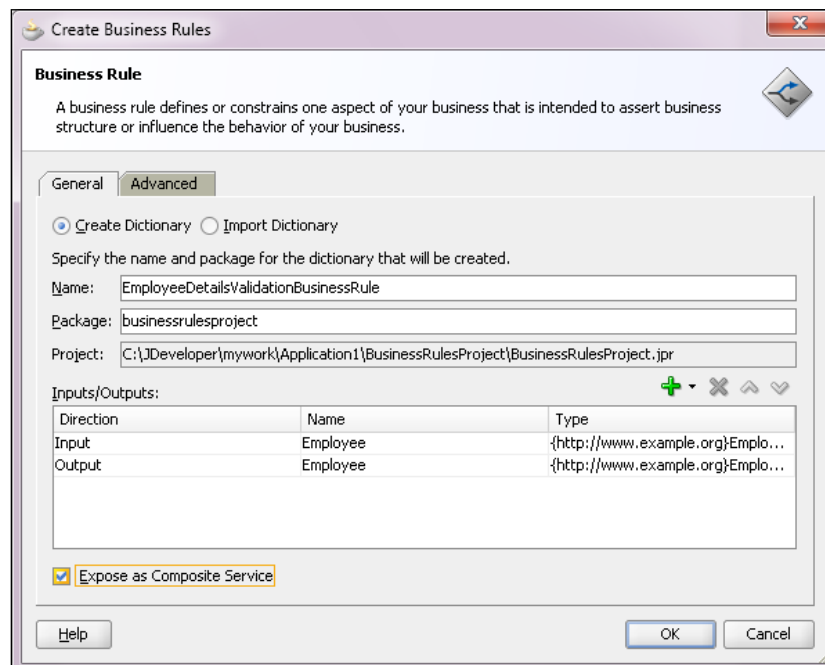


2. Create a new Dictionary with input and output from the Business Rules, as shown in the following screenshots:



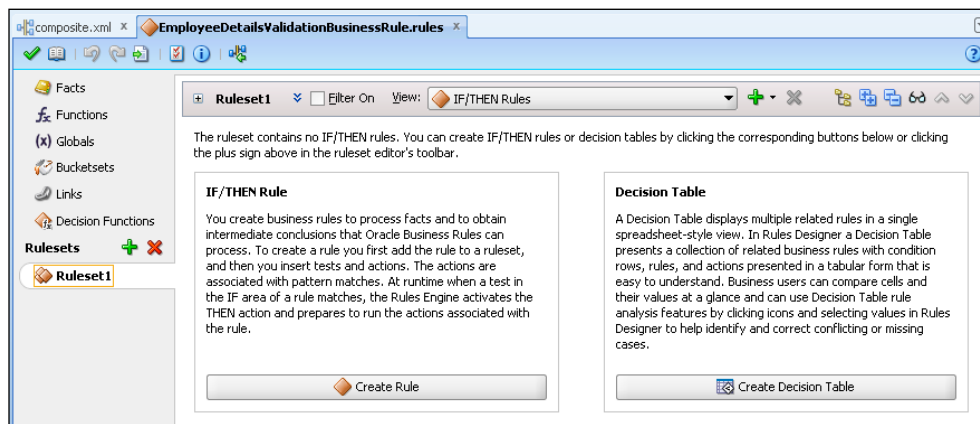


3. Expose the Business Rule as a service accessible to others as follows:

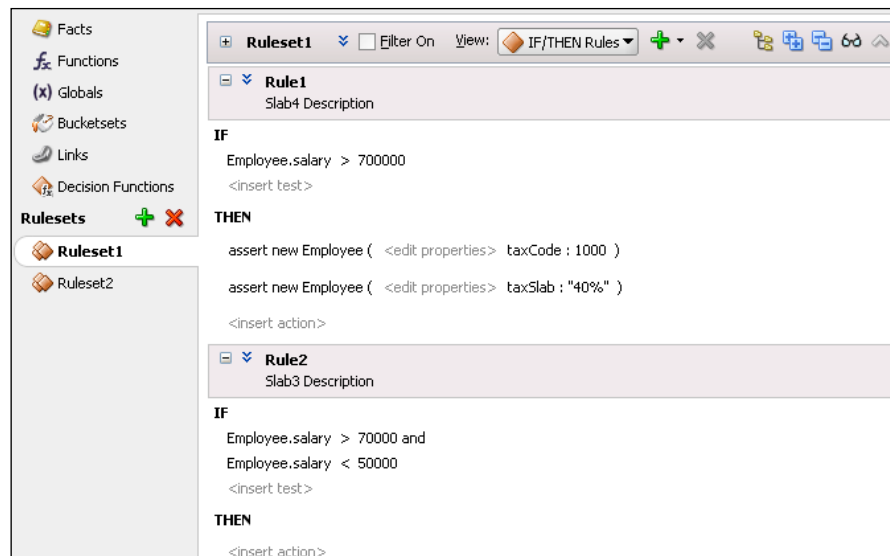




4. Choose the style of the Business Rule implementation as follows:



An example of the if-then rule is shown as follows:



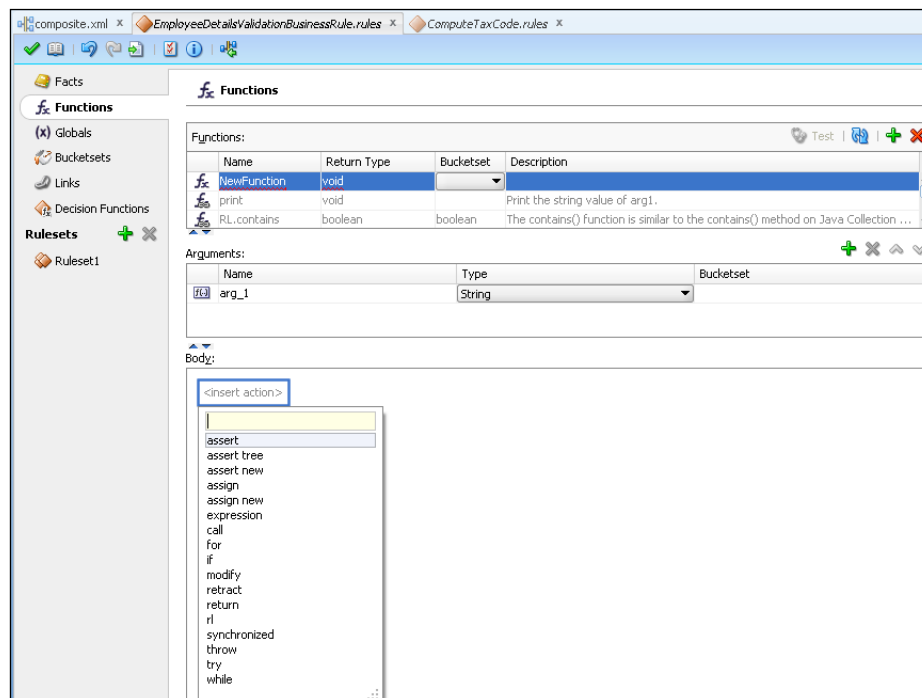


An example of the Decision Table is shown as follows:

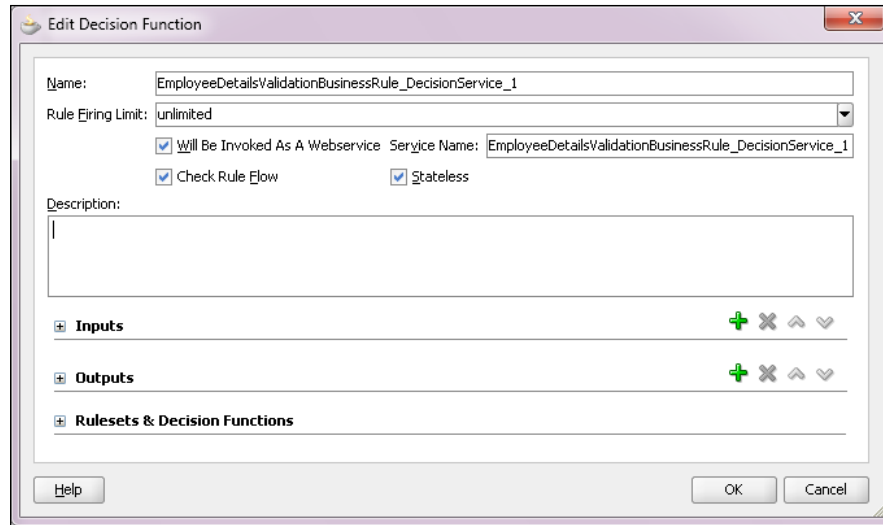
The screenshot shows the Oracle Business Rules interface. On the left is a sidebar with a tree view containing 'Facts', 'Functions', 'Globals', 'Bucketsets', 'Links', 'Decision Functions', and 'Rulesets'. Under 'Rulesets', 'Ruleset1' and 'Ruleset2' are listed, with 'Ruleset2' selected. The main area displays 'Ruleset2' with a view dropdown set to 'DecisionTableToEvaluate Rules'. The table has four columns: 'Conditions', 'R1', 'R2', 'R3', and 'R4'. The first row, 'C1 Employee.salary', contains the conditions '<20000', '[20000..30000)', '[30000..40000)', and '>=40000'. Below this is a 'Conflict Resolution' section. The 'Actions' section contains 'A1 assert new Employee(' with parameters 'taxCode: Object' and 'taxSlab: Object'. The actions are mapped to the rules: R1 has '5000' and '10%', R2 has '6000' and '20%', R3 has '7000' and '30%', and R4 has '8000' and '40%'. Each action cell has a checkbox.

Conditions	R1	R2	R3	R4
C1 Employee.salary	<20000	[20000..30000)	[30000..40000)	>=40000
<b>Conflict Resolution</b>				
<b>Actions</b>				
A1 assert new Employee( taxCode: Object taxSlab: Object)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	5000	6000	7000	8000
	"10%"	"20%"	"30%"	"40%"

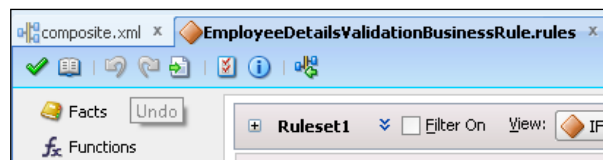
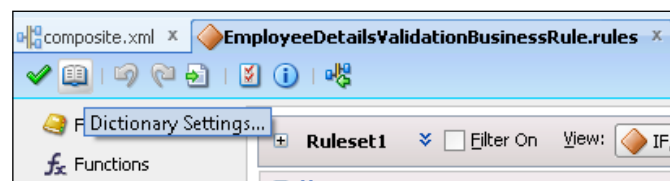
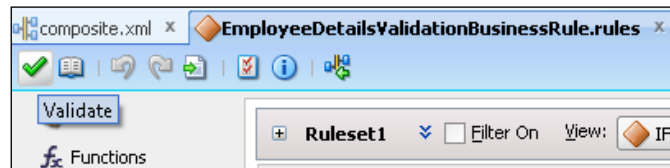
The **Custom Business Rules Function** can be defined using the **Functions** tab in Oracle Business Rules as follows:

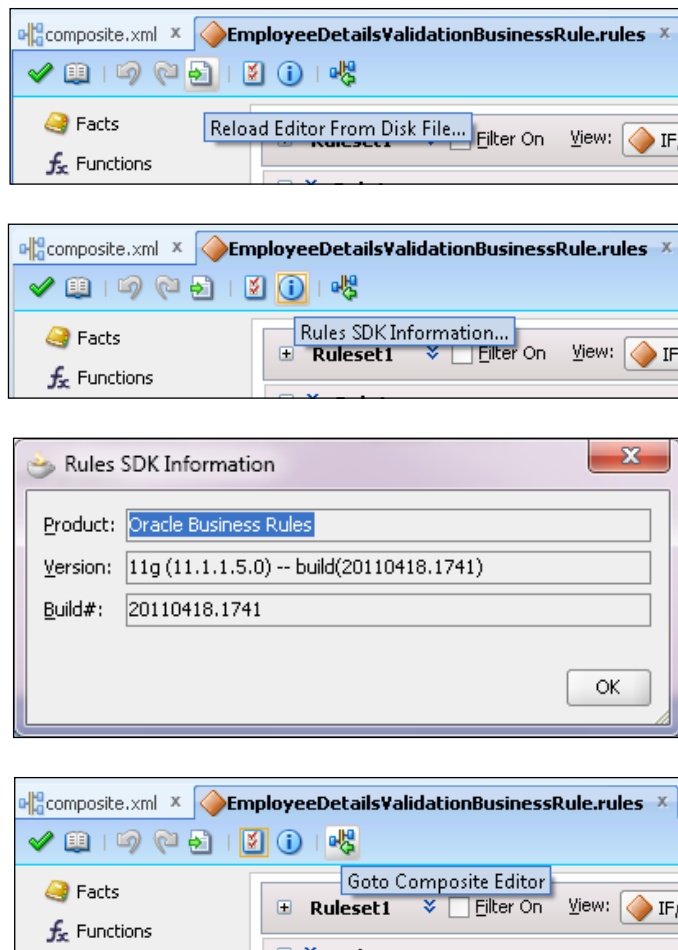


**Decision Functions** provide the features to set the **Rule Firing Limit**, **State Definition**, **Service Invocation** feature, and select the RuleSets that will be part of the **Decision Services** to be accessed from Java, service components, or from any soap-clients as Web Services:



JDeveloper provides options to Validate, Undo, Redo, identify Rule Dictionary Information, and go to the associated Composite Editor through the icons highlighted in the following screenshots:





## Integrating simple Business Rules with the BPEL process

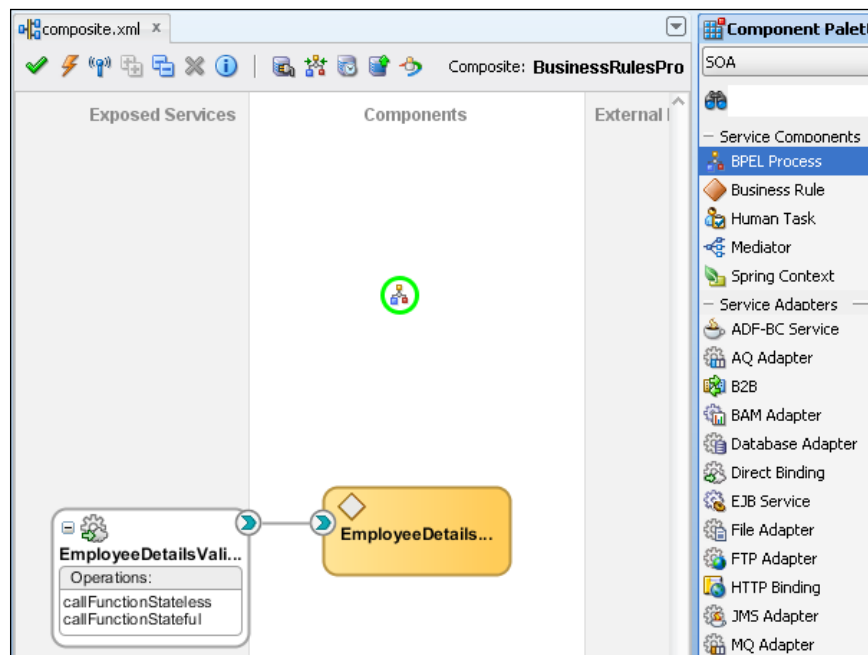
Business Rules can be integrated with Business Rules in two fashions. This section of the book will explore both the options specified previously:

- Wiring the Business Rule component through SOA Composite
- Including the Business Rule as an activity to the BPEL process

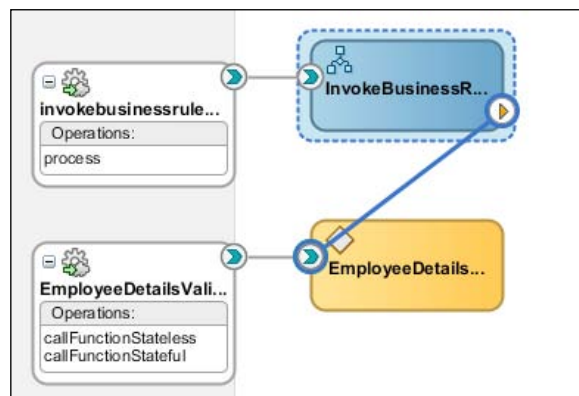
## Wiring the Business Rule component through SOA Composite

The steps for wiring the Business Rule component through SOA Composite are as follows:

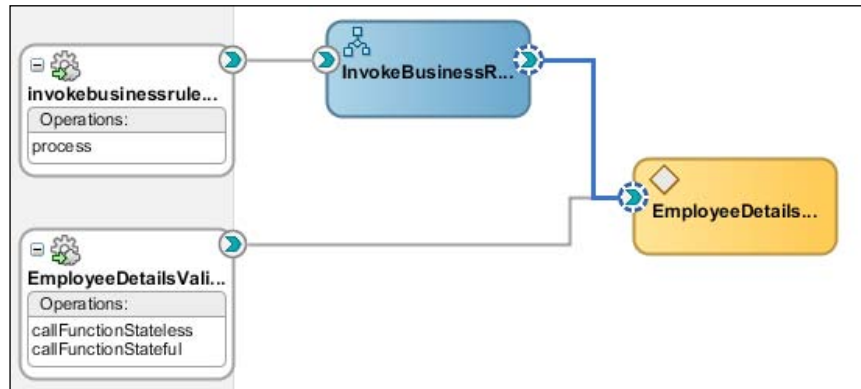
1. Drag-and-drop the BPEL process:



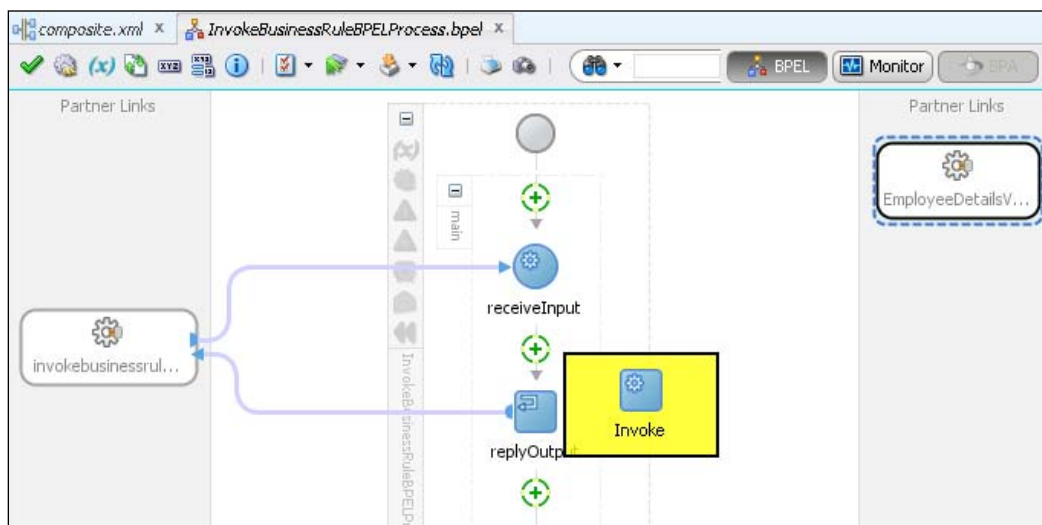
2. Invoke the Business Rule from BPEL through wire:



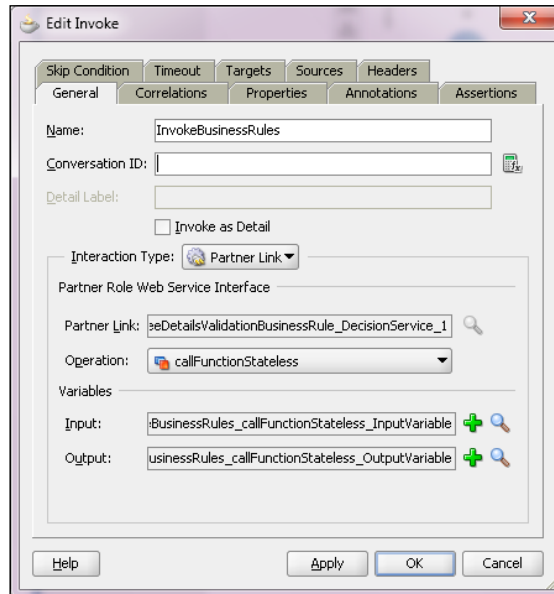
A connection should be established as follows:



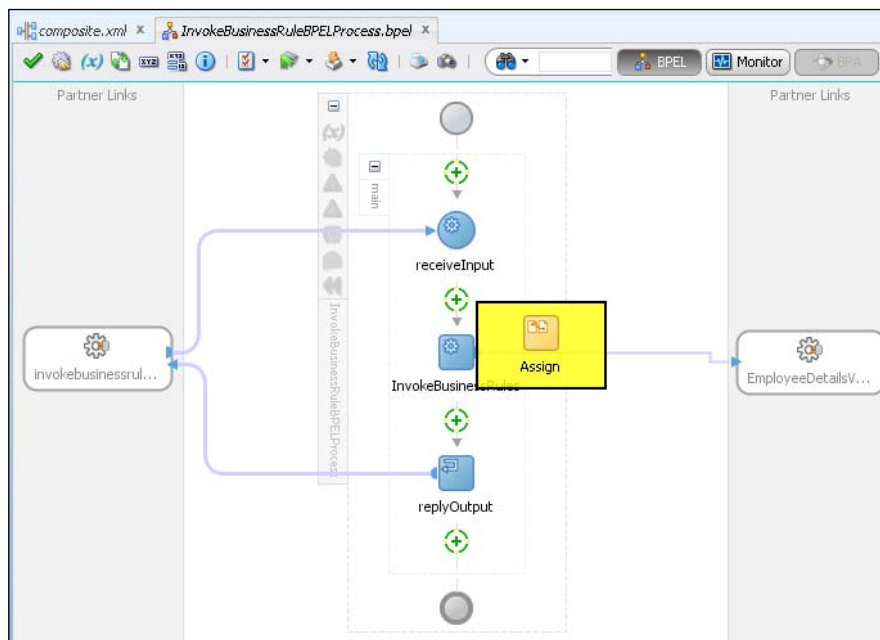
3. Open the BPEL process and add the **Invoke** activity as follows:



4. Add the Input and Output variables to the **Invoke** activity:



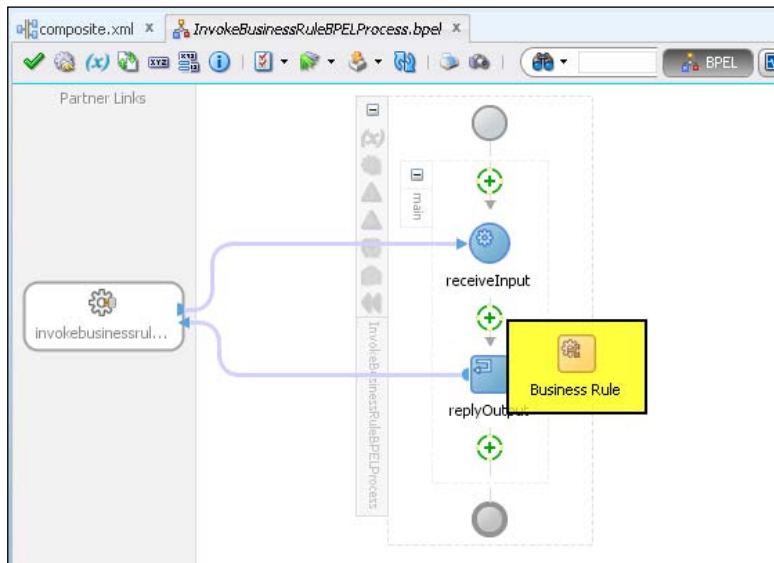
5. Add an **Assign** activity to populate the Input variable to the Business Rules as follows:



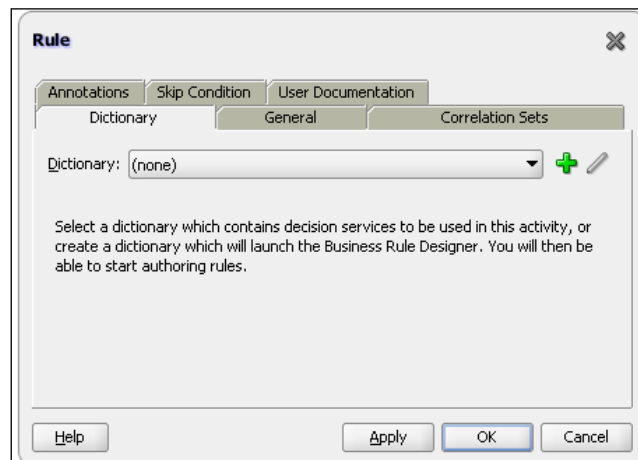
## Including the Business Rule as an activity to the BPEL process

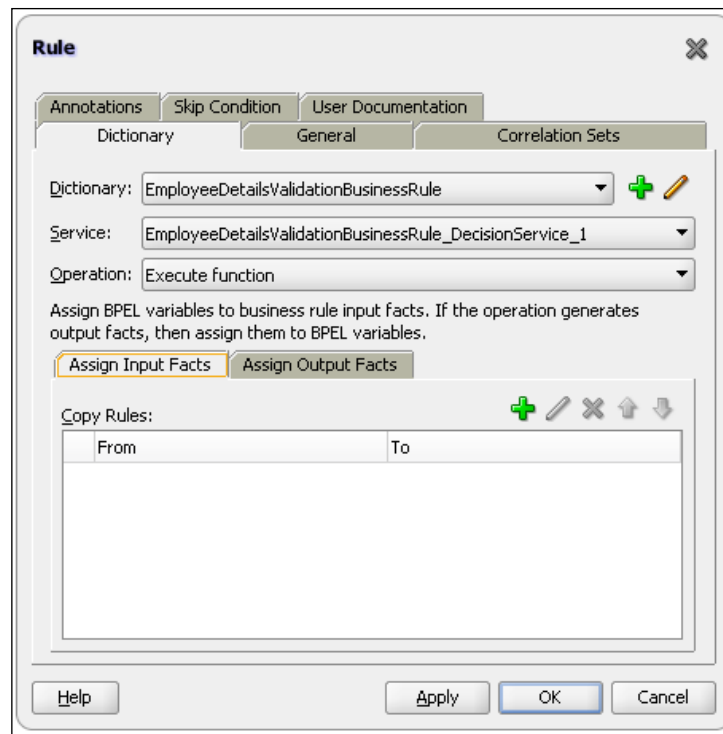
The steps for including the Business Rule as an activity to the BPEL process are as follows:

1. Open the BPEL process and add the Business Rule activity, as shown in the following image:



2. Double-click on the Business Rules and choose **Dictionary**:





## Testing Business Rules from JDeveloper

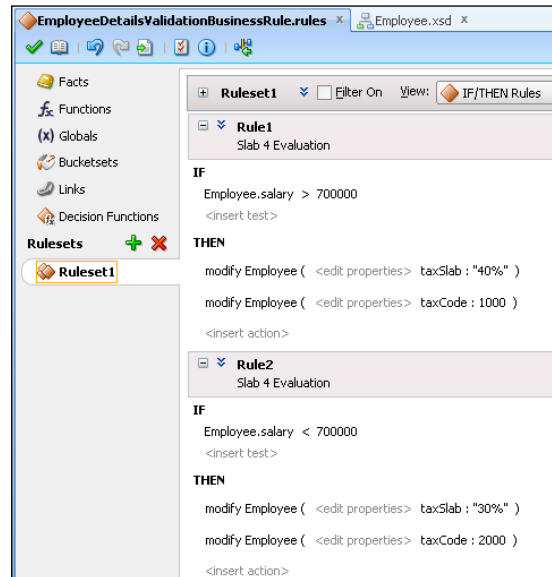
Testing Business Rules can be done from both JDeveloper and SOA EM console. This section will describe the technique that can be used to test the Business Rule from JDeveloper.

To test the Business Rule from JDeveloper, one should write a Test function and assert/assign the input value to see the results.

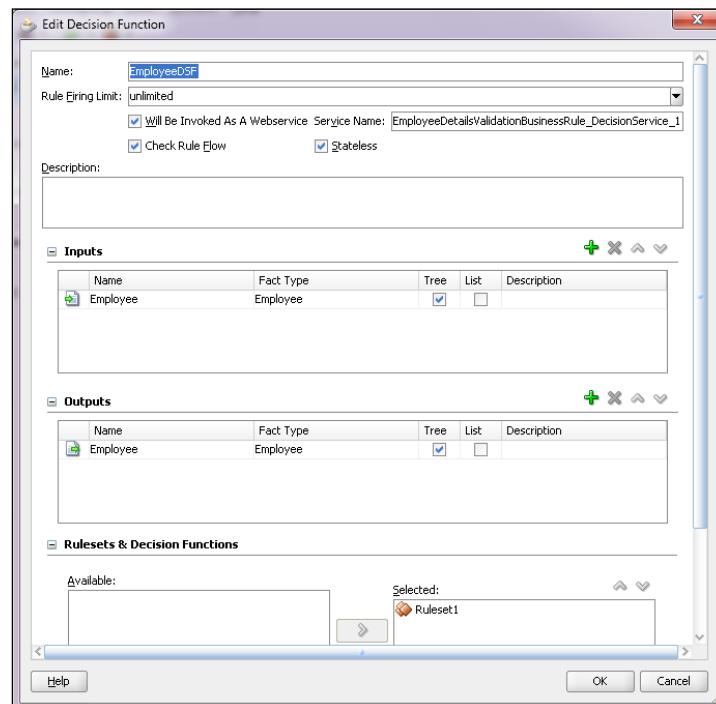
The following screenshots describe the Business Rules setup and Testing function details:



The rule details are shown as follows:



The Decision Function setup details are shown as follows:



**fx Functions**

Functions: Test + ×

	Name	Return Type	Bucketset	Description
	TestBusinessRule	boolean	boolean	
	print	void		Print the string value of arg1.
	RL.contains	boolean	boolean	The contains() function is similar to the contains() method on Java Collection but includ...

Arguments: + × ↕ ↩

Name	Type	Bucketset

Body:

```

assign new Employee var = new Employee()

assign var.employeeID = "1"

assign var.age = "27"

assign var.taxSlab = "10%"

assign var.taxCode = "0"

assign var.sex = "Male"

assign var.salary = 30000

assign new List var1 = EmployeeDSF(var)

assign new Employee var2 = new Employee()

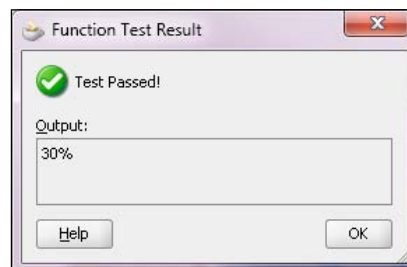
assign var2 = (Employee) var1.get(0)

call print( message : var2.taxSlab )

return true

<insert action>

```

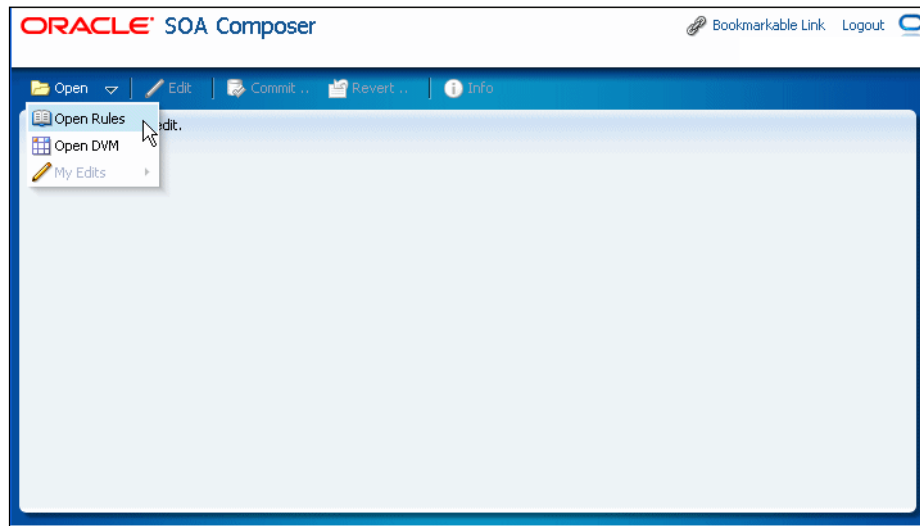


## Editing Business Rules from SOA Composer

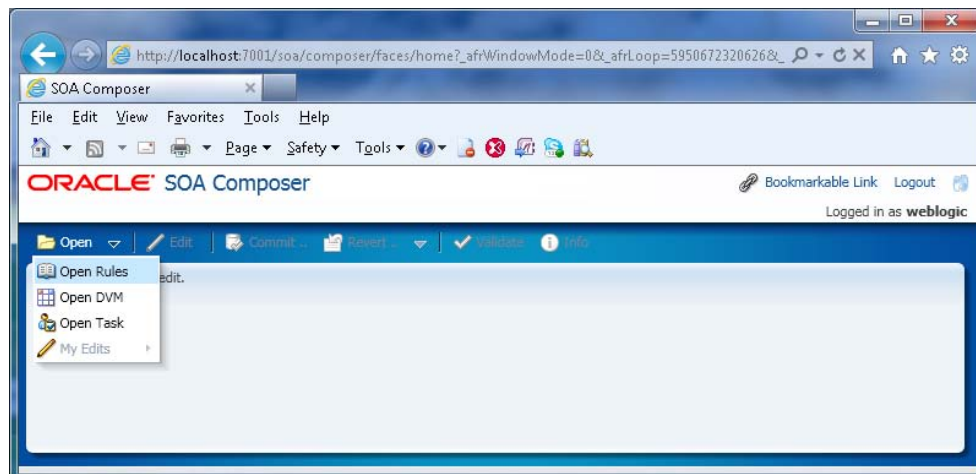
Oracle provides the features to edit the Oracle SOA Business Rules through SOA Composer.

SOA Composer can be launched through the following URL:

`http://<localhost>:<default_port>/soa/composer`



The dialog to open the existing Business Rule is as follows:



---

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Explain the Business Rules concepts:** We introduced the need and significance of Business Rules in this section
- **Describe the Oracle Business Rules Architecture and Oracle Rules Engine details:** We discussed about the RETE algorithm and Oracle Business Rules Architecture
- **Create rules with the JDeveloper Rules Designer:** We identified various UI features available to create and test business rules from JDeveloper
- **Integrate a simple rule with a BPEL process:** This section identified the steps to integrate rules with a BPEL process

## Self-review questions

1. Oracle Business Rule is a data-driven \_\_\_\_\_ chaining system.
  - a. Rule
  - b. Forward
  - c. Rule-based
  - d. Backward
2. Oracle Business Rule provides a safe feature to edit the rules by multiple users.
  - a. True
  - b. False
3. Oracle Business Rule Engine provides the features to change the rules without stopping the Business Process implementing.
  - a. True
  - b. False
4. Oracle Business Rule-based Engine consists of.
  - a. Rule-Bases
  - b. Working Memory
  - c. Inference engine
  - d. All the above

5. Oracle Business Rules \_\_\_\_\_ provide a contract to access from Java or SOA Composite.
  - a. Facts
  - b. Ruleset
  - c. Decision Functions
  - d. Dictionaries
6. Oracle Business Rules contain a \_\_\_\_\_.
  - a. Rule Dictionary
  - b. Rule Engine
  - c. Rule Library
  - d. Rule Editor
7. Identify the statement that describes the Rules Component of Oracle Business Rules.
  - a. Rules are data or business objects on which the rules engine evaluates rule conditions.
  - b. Rules are declared as "if condition then action".
  - c. Rules have an action to assign, assert, or call function (or a Java method).
  - d. Rules have Decision Tables to evaluate the rule conditions.
8. Identify the examples of business rules.
  - a. If the age of a customer is less than 18, then decline to sell.
  - b. If a traveler account has total miles for the year greater than 500,000, then their status is Platinum.
  - c. If the total annual income of the family is less than \$25,000, then deny the loan.
  - d. All the above

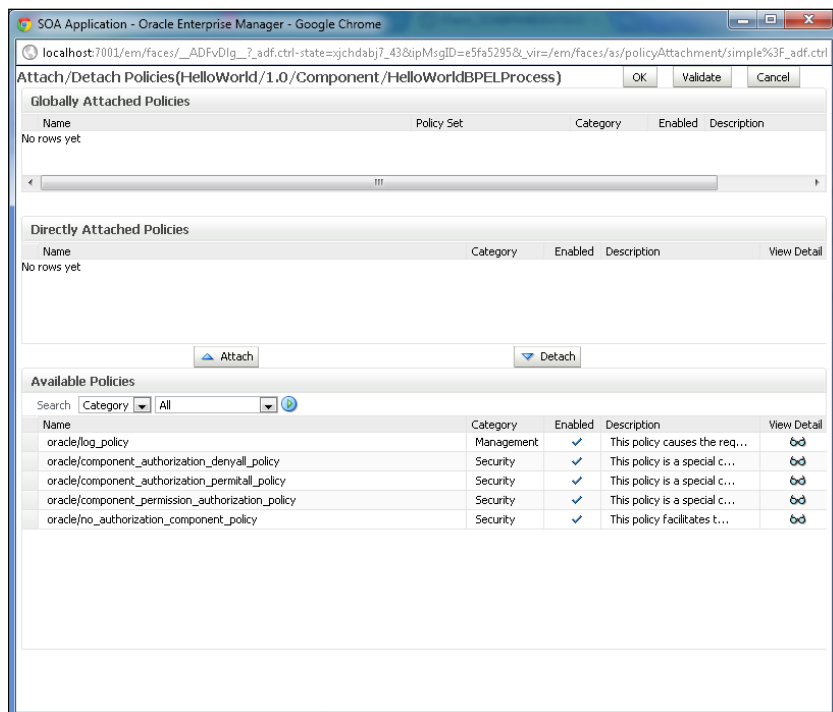
9. The Business Rule Dictionary is stored in a file with \_\_\_\_\_ extension.
- a. \*.rules
  - b. \*.decs
  - c. \*.dict
  - d. \*.dictionaries
10. List of values can be defined using:
- a. Facts
  - b. Bucketsets
  - c. Links
  - d. Decision Functions



# 10

## Securing Services and Composite Applications

Securing computer resource is a key objective for any enterprise application component. Web Services are no exception to this. Oracle SOA Composite Application provides features to attach and detach WS-Policies in a configurable style to manage the security components of the Web Services. An example of the SOA Application Policy configuration is shown as follows:





This chapter of the book will cover the following exam objectives:

- Explain Web Services and composite security
- Describe the Fusion Middleware security architecture
- Understand security policy management and identity propagation
- Describe OWSM Agents

## Web Services and composite security

Web Services' security requirements involve credential transfer and service capabilities along with defining application constraints. In many cases, Web Services security tools, such as Oracle WSM, rely on **Public Key Infrastructure (PKI)** environments. A PKI uses cryptographic keys. Keys can be private or public. In an asymmetric cipher model, the receiving party's public key is used to encrypt plaintext, and the receiving party's matching private key is used to decrypt the ciphertext. Also, a private key is used to create a digital signature by signing the message and the public key is used for verifying the signature. Public-key certificates (or certificates, for short) are used to guarantee the integrity of public keys.

Web Services' security requirements are supported by industry standards both at the transport level (Secure Socket Layer) and at the application level, relying on XML frameworks.

Oracle Web Services Manager is designed to define and implement Web Services' security in heterogeneous environments, including authentication, authorization, message encryption and decryption, signature generation and validation, and identity propagation across multiple Web Services.

Security standards are implemented in non-XML frameworks at the transport level and in XML frameworks at the application level.

The following sections describe the standards that are key to providing secure and manageable SOA environments at both the transport and application levels:

- Transport Layer Security – SSL
- XML encryption (data confidentiality)
- XML signature (data integrity, data authenticity)
- WS-Security
- WS-Security tokens
- WS-Policy
- WS-SecurityPolicy

- Web Services Addressing (WS-Addressing)
- WS-ReliableMessaging

## Transport-level security

**Transport Layer Security (TLS)** is a SSL protocol officially standardized by **Internet Engineering Task Force (IETF)**. It is the most widely used transport-level data-communication protocol providing:

- Authentication confidentiality message integrity
- Secure key exchange between client and server

SSL provides point-to-point security. SSL provides a secure communication channel. However, when the data is not in transit, the data is not protected. This makes the communication vulnerable to different sets of attacks in a multi-step transactions scenario.

SSL can be used in three modes:

- No authentication
- One-way authentication (server authentication)
- Two-way authentication (bilateral authentication)

SSL uses a combination of secret-key and public-key cryptography to secure communications. SSL traffic uses secret keys for encryption and decryption, and the exchange of public keys is used for mutual authentication of the parties involved in the communication.

## Application-level security

Application-level security complements transport-level security. Application-level security is based on XML frameworks defining confidentiality, integrity, authenticity, message structure, trust management, and federation.

Data confidentiality is implemented by XML Encryption. XML Encryption defines how digital content is encrypted and decrypted, how the encryption key information is passed to a recipient, and how encrypted data is identified to facilitate decryption.

SAML is one of the most interesting security tokens because it supports both authentication and authorization. **SAML** is an open framework for sharing security information on the Internet through XML documents.

The full SAML specification is used in browser-based federation cases. However, Web Services' security systems such as Oracle WSM only use SAML assertions. The protocol and bindings are taken care of by WS-Security and the transport protocol.

SAML assertions and references to assertion identifiers are contained in the WS-Security Header element, which in turn is included in the SOAP Envelope Header element. The SAML security token is particularly relevant in situations where identity propagation is essential.

## **XML Encryption (data confidentiality)**

The **XML Encryption** specification describes a process for encrypting data and representing the result in XML. Specifically, XML Encryption defines:

- How to encrypt and decrypt the digital content
- How the encryption key information is sent to a recipient
- How the encrypted data is identified

An XML document may be encrypted as a whole or in part.

An XML representation of some credit card data is shown as follows:

```
<PaymentInfo xmlns="http://www.example.com/payment">
  <CreditCard>
    <Name>John Smith</Name>
    <CreditCardNumber>2222 4444 7777 1234</NCreditCardNumber>
    <Limit>30000</Limit>
    <Issuer>Example Bank</Issuer>
    <Expiration>07/20</Expiration>
  </CreditCard>
</PaymentInfo>
```

An XML representation of encrypted credit card data is shown as follows:

```
<PaymentInfo xmlns='http://www.example.com/payment">
  <CreditCard>
    <Name>John Smith</Name>
    <CreditcardNumber>
      <EncryptedData xmlns="http://www..." Type="http://www...">
        <CipherData>
          <CipherValue>A123DECB4...5EDD5EF6</CipherValue>
        </CipherData>
      </EncryptedData>
    <Limit>30000</Limit>
    <Issuer>Example Bank</Issuer>
    <Expiration>07/20</Expiration>
  </CreditCard>
</PaymentInfo>
```

## XML Signature

The **XML Signature** specification describes signature processing rules. XML signature binds the sender's identity to an XML document. The document is signed using the sender's private key; the signature is verified using the sender's public key.

Signing and signature verification can be done using asymmetric or symmetric keys. An XML Signature also ensures non-repudiation of the signing entity, that is, it provides proof that messages have not been altered since they were signed.

A signature can apply to a whole document or just to a part of a document.

## WS-Security

**Web Services Security (WS-Security)** specifies SOAP security extensions that provide confidentiality using XML encryption and data integrity using an XML signature. WS-Security also includes profiles that specify how to insert different types of binary and XML security tokens in WS-Security headers for authentication and authorisation purposes. WS-Security token profiles are described in the following sections.

## WS-Security tokens

Web Services security supports the following security tokens:

- Username – It defines how a Web Service consumer should supply a username for authentication
- X.509 certificate – Defines the signed data structure designed to send a public key to a receiving party
- Security Assertion Markup Language (SAML) assertion – A standard that supports security information over the Internet through XML documents
- Kerberos ticket provides the binary authentication and session token

## Username

The **username** token carries basic authentication information. The username token element propagates username and password information to authenticate the message. The information provided in the token and the trust relationship provide the basis for establishing the identity of the user.

## **X.509 certificate**

An **X.509 digital certificate** is a signed data structure designed to send a public key to a receiving party. A certificate includes standard fields such as certificate ID, issuer's Distinguished Name (DN), validity period, owner's DN, owner's public key, and so on.

Certificates are issued by certificate authorities (CA). A CA verifies an entity's identity and grants a certificate, signing it with the CA's private key. The CA publishes its own certificate, which includes its public key.

Each network entity has a list of the certificates of the CAs it trusts. Before communicating with another entity, a given entity uses this list to verify that the signature of the other entity's certificate is from a trusted CA.

## **Kerberos ticket**

**Kerberos** is a cross-platform authentication and a single sign-on system. The Kerberos protocol provides mutual authentication between two entities relying on a shared secret (symmetric keys). Kerberos uses the following terminology:

- A **principal** is an identity for a user (that is, a user is assigned a principal), or an identity for an application offering Kerberos services.
- A **realm** is a Kerberos server environment; a Kerberos realm can be a domain name such as sample.com (by convention, expressed in uppercase).

Kerberos involves a client, a server, and a trusted party to mediate between them called the **Key Distribution Center (KDC)**. Each Kerberos realm has at least one KDC. KDCs come in different packages based on the operating platform used. The Kerberos token profile of WS-Security allows business partners to use Kerberos tokens in service-oriented architectures.

## **SAML token**

The **Security Assertion Markup Language (SAML)** is an open standard framework for sharing security information over the Internet through XML documents. SAML was designed to address the following:

- **Web Services security:** SAML provides a standard security token that can be used with standard Web Services security frameworks. This is the use of SAML, which is particularly relevant to Web Services security and fully supported by Oracle WSM.

- **Identity propagation:** SAML provides a standard way to represent a security token that can be passed across the multiple steps of a business process or transaction from browser to portal to networks of web services, which is also a feature supported by Oracle WSM.
- **Limitations of web browser cookies:** SAML provides a standard way to transfer cookies across multiple Internet domains.
- **Proprietary web single sign-on (SSO):** SAML provides a standard way to implement SSO within a single domain or across multiple domains. This functionality is provided by the Oracle Identity Federation product.
- **Federation:** SAML facilitates identity management through account linking when a single user is known to multiple websites under different user identities, also supported by Oracle Identity Federation.

The SAML framework includes four parts:

- **Assertions:** How you define authentication and authorization information
- **Protocols:** How you ask SAML request and get SAML response the assertions you need
- **Bindings:** How SAML protocols ride on industry-standard transport and messaging frameworks
- **Profiles:** How SAML protocols and bindings combine to support specific use cases

SAML assertions are very popular security tokens within WS-Security because they are very expressive and can help prevent man-in-the-middle and replay attacks.

Typically, an SAML assertion makes statements about a principal (a user or an application). All SAML assertions include the following common information:

- Issuer ID and issuance timestamp
- Assertion ID
- Subject
- Name
- Optional subject confirmation
- Optional conditions
- Optional advice

## **WS-Policy**

WS-Policy is another key industry standard for Oracle Fusion Middleware security.

A Web Service provider may define conditions under which a service is to be provided. The WS-Policy framework enables one to specify policy information that can be processed by Web Services applications such as Oracle WSM.

A policy is expressed as one or more policy assertions representing a Web Service's capabilities or requirements. For example, a policy assertion may stipulate that a request to a Web Service be encrypted. Likewise, a policy assertion can define the maximum message size that a Web Service can accept.

WS-Policy expressions are associated with various Web Services components using the WS-Policy attachment specification. WS-Policy information can be embedded in a WSDL file, thus making it easy to expose Web Service policies through a UDDI registry.

## **WS-SecurityPolicy**

WS-SecurityPolicy is part of the Web Services Secure Exchange set of specifications hosted by OASIS.

**WS-SecurityPolicy** defines a set of security policy assertions used in the context of the WS-Policy framework. WS-SecurityPolicy assertions describe how messages are secured on a communication path. Oracle has contributed to the OASIS WS-SX technical committee for several practical security scenarios. Each security scenario describes WS-SecurityPolicy policy expressions.

WS-SecurityPolicy scenarios describe examples of how to set up WS-SecurityPolicy policies for several security token types, described in the WS-Security specification. The subset of the WS-SecurityPolicy scenarios supported by Oracle WSM 11g represents the most common customer use cases. Each scenario has been tested in multiple-vendor WS-Security environments.

To illustrate the WS-SecurityPolicy, let's use a scenario supported by Oracle WSM, UsernameToken, with a plain text password. As mentioned earlier, the username token is one of the security tokens specified by WS-Security. This specific scenario uses a policy that says that a requester must send a password in a username token to a recipient who has the authority to validate that token. The password is a default requirement for the WS-Security Username Token Profile 1.1. This scenario is only recommended when confidentiality of the password is not an issue, such as a pre-production test scenario with dummy passwords.

The WS-SecurityPolicy structure is shown as follows:

```
<wsp:Policy>
  <sp:SupportingTokens>
    <wsp:Policy>
      <sp:UsernameToken/>
    </wsp:Policy>
  </sp:SupportingTokens>
</wsp:Policy>
```

An example of a message that conforms to the previously stated policy is shown as follows:

A message conforming to WS-SecurityPolicy is shown as follows:

```
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope xmlns:soap="...">
  <soap:Header>
    <wsse:Security soap:mustUnderstand="1" xmlns:wsse="...">
      <wsse:UsernameToken>
        <wsse:Username>Sam</wsse:Username>
        <wsse:Password Type="http://docs.oasis open.org...">
          ABCD
        </wsse:Password>
        <wsse:Nonce EncodingType="...#Base64Binary">qB...</wsse:Nonce>
        <wsu:Created>2012-01-02T00:01:03Z</wsu:Created>
      </wsse:UsernameToken>
    </wsse:Security>
  </soap:Header>
  <soap:Body>
    <Oracle xmlns=http://xmlsoap.org/Oracle>
      <text>EchoString</text>
    </Oracle>
  </soap:Body>
</soap:Envelope>
```

The previous example contains a <Nonce> element and a <Created> timestamp, which, while optional, are recommended to improve security of requests against replay and other attacks. A **nonce** is a randomly generated (unique) number. The timestamp can be used to define the amount of time the security token is valid.



## Web Services Addressing (WS-Addressing)

SOAP does not provide a standard way to specify where a message is going or how responses or faults are returned. **WS-Addressing** provides an XML framework for identifying Web Services endpoints and for securing end-to-end endpoint identification in messages.

WS-Addressing is transport-independent; that is, the request may be over JMS and the response over HTTP. WS-Addressing is used with other WS-\* specifications, such as WS-Policy.

## WS-ReliableMessaging

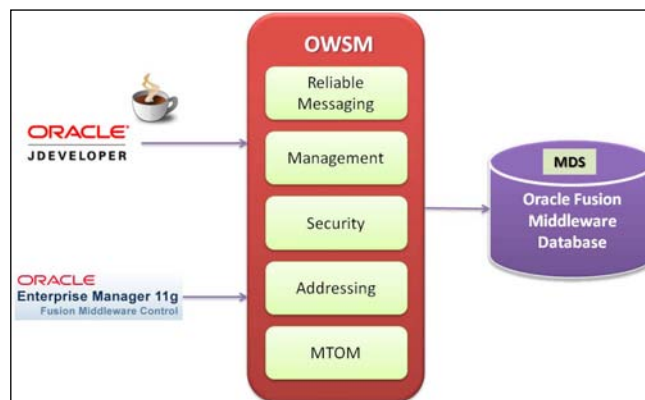
**WS-ReliableMessaging (WS-RM)** defines a framework for identifying and managing the reliable delivery of messages between Web Services endpoints. WS-RM is predicated on the SOAP messaging structure and relies on WS-Security, WS-Policy, and WS-Addressing to provide reliable messaging.

WS-RM defines a reliable messaging (RM) source and an RM destination. WS-RM mandates prerequisites, for example, trust between endpoints must be established and the message and endpoints must be formally identified.

WS-RM policy defines a policy assertion that leverages the WS-Policy framework in order to enable an RM destination and an RM source to describe their requirements for a given sequence.

## The Fusion Middleware Security architecture

The Fusion Middleware Security architecture is shown in the following image:



Oracle WSM Component	Description
Oracle Enterprise Manager Fusion Middleware Control	Enables administrators to access Oracle WSM's functionality to manage, secure, and monitor Web Services.
Oracle WSM Policy Manager	Reads/writes the policies, including predefined and custom policies from the metadata store.
Oracle WSM Agent	Manages the enforcement of policies via the Policy Interceptor Pipeline.
Policy Interceptors	Enforces policies, including reliable messaging, management, addressing, security, and Message Transmission Optimization Mechanism (MTOM).
Metadata Store (MDS)	Stores policies. Policies can be stored either as files in the filesystem (supported for development) or to the Oracle Fusion Middleware database (supported for production).
Oracle Fusion Middleware Database	Provides database support for the MDS.

There are two types of policies that can be attached to Web Services: Oracle WSM policies and WebLogic Server policies.



Note: Policies that are attached to WebLogic Web Services at design time cannot be detached at deployment time. You can only attach new policies.

## Security policy management and identity propagation

Policies describe the capabilities and requirements of a Web Service such as whether and how a message must be secured, whether and how a message must be delivered reliably, and so on.

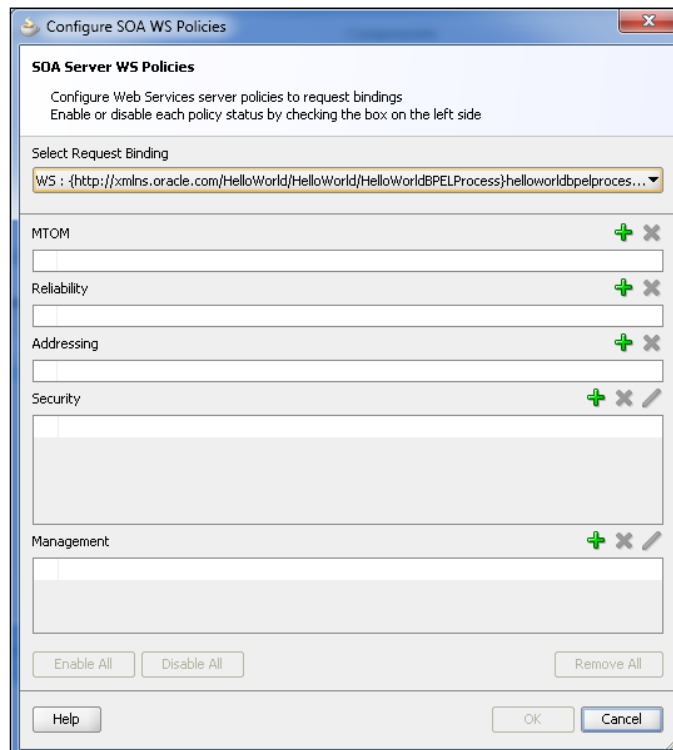
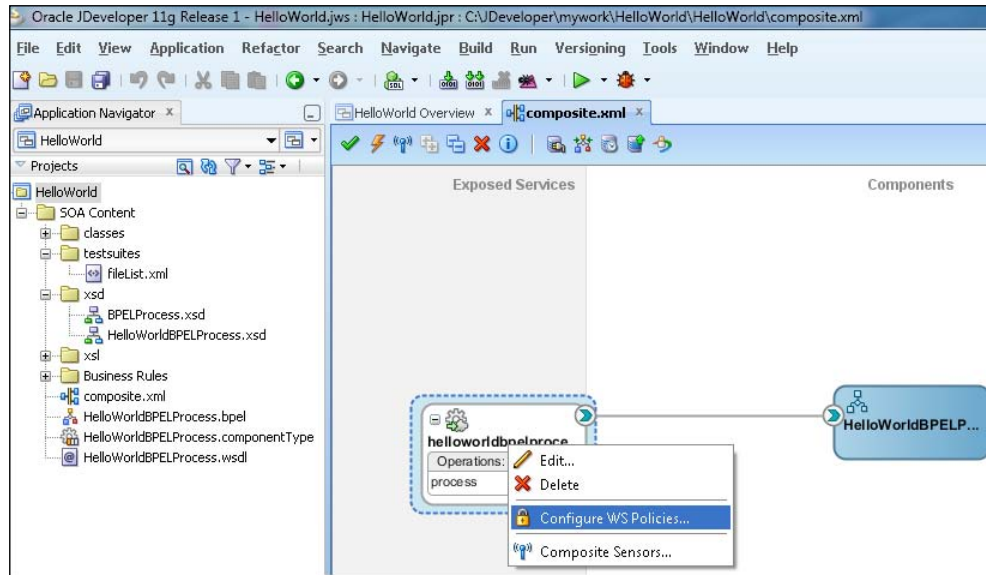
Oracle Fusion Middleware 11g Release 1 (11.1.1) supports the following types of policies:

- **WS-ReliableMessaging** – Reliable messaging policies that implement the WS-ReliableMessaging standard describe a wire-level protocol that allows guaranteed delivery of SOAP messages and can maintain the order of the sequence in which a set of messages are delivered.

The technology can be used to ensure that messages are delivered in the correct order. If a message is delivered out of order, the receiving system can be configured to guarantee that the messages will be processed in the correct order. The system can also be configured to deliver messages at least once, not more than once, or exactly once. If a message is lost, the sending system re-transmits the message until the receiving system acknowledges its reception.

- **Management** – Management policies log request, response, and fault messages to a message log. Management policies may include custom policies.
- **WS-Addressing** – WS-Addressing policies that verify that SOAP messages include WS-Addressing headers in conformance with the WS-Addressing specification. Transport-level data is included in the XML message rather than relying on the network-level transport to convey this information.
- **Security** – Security policies implement the WS-Security 1.0 and 1.1 standards. They enforce message protection (message integrity and message confidentiality) and authentication and authorization of Web Service requesters and providers. The following token profiles are supported:
  - username token
  - X.509 certificate
  - Kerberos ticket
  - Security Assertion Markup Language (SAML) assertion.
- **Message Transmission Optimization Mechanism (MTOM)** – Binary content, such as an image in JPEG format, can be passed between the client and the Web Service. In order to be passed, the binary content is typically inserted into an XML document as an `xsd:base64Binary` string. Transmitting the binary content in this format greatly increases the size of the message sent over the wire and is expensive in terms of the required processing space and time.
- Using Message Transmission Optimization Mechanism (MTOM), binary content can be sent as a MIME attachment, which reduces the transmission size on the wire. The binary content is semantically part of the XML document. Attaching an MTOM policy ensures that the message is converted to a MIME attachment before it is sent to the Web Service or client.
- The policies are part of the Oracle WSM enterprise policy framework, which allows policies to be centrally created and managed.

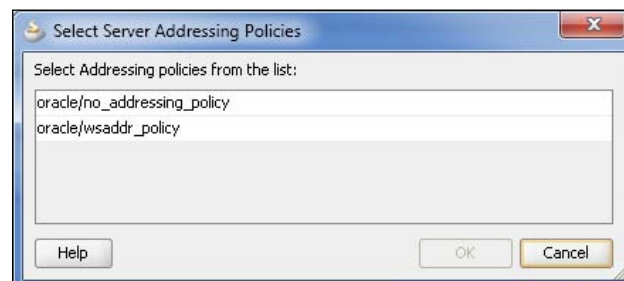
The following screenshots show the WS-Policy configuration steps:



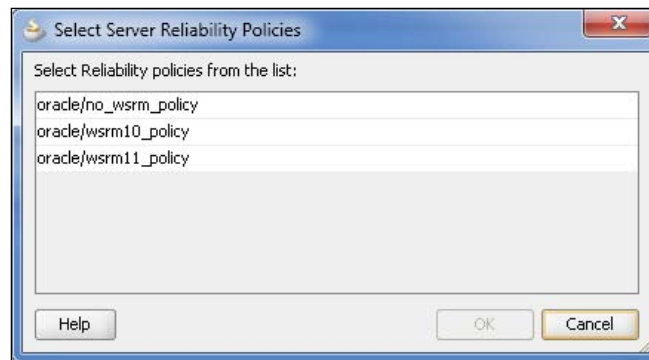
The following screenshot shows the list of security policies available from Oracle:



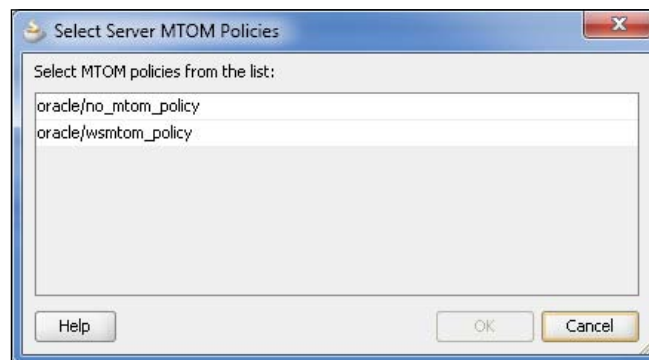
The following screenshot shows the list of WS-Addressing policies available from Oracle:



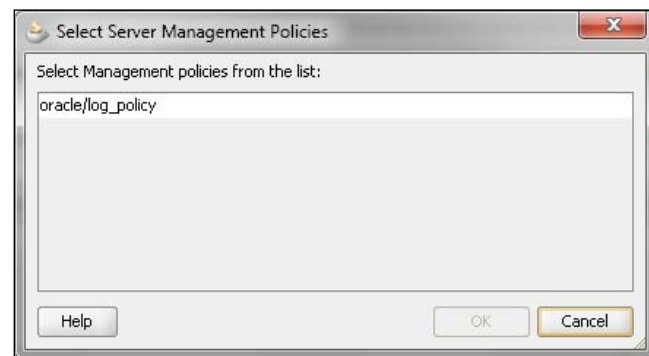
The following screenshot shows the list of WS-Reliability policies available from Oracle:



The following screenshot shows the list of MTOM policies available from Oracle:



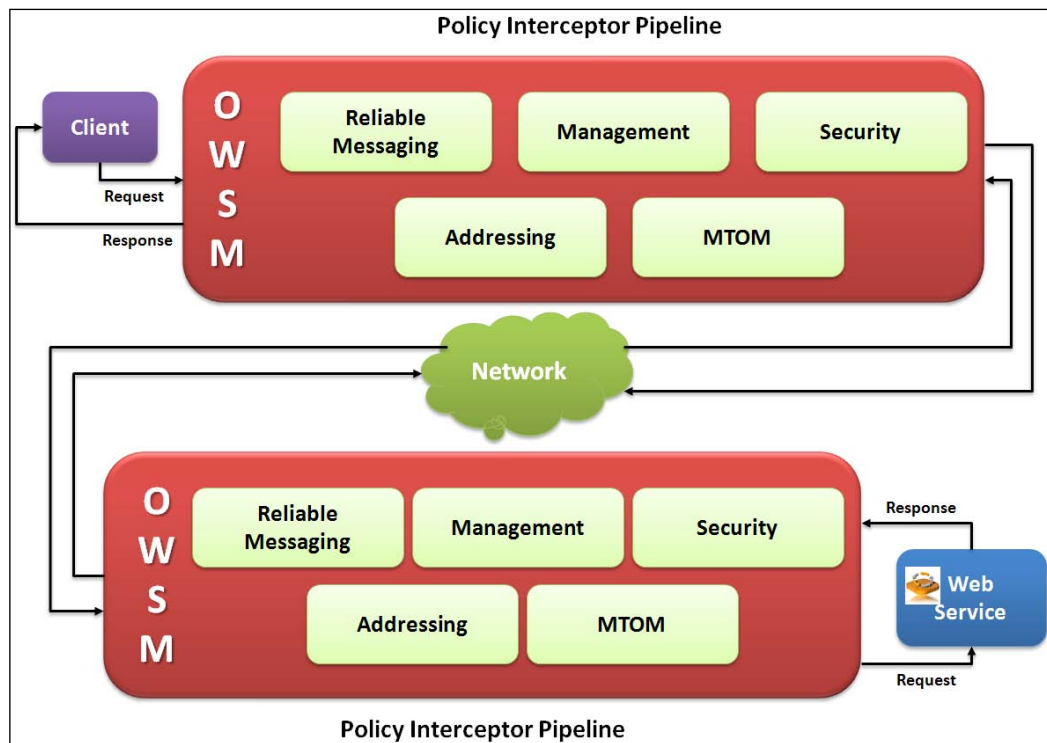
The following screenshot shows the list of Management policies available from Oracle:



## OWSM Agents

OWSM manages the enforcement of policies via the Policy Interceptor Pipeline. When a request is made from a service consumer to a service provider, the request is intercepted by one or more policy interceptors. These interceptors execute policies that are attached to the client and to the Web Service. Each interceptor executes policies of the same type. For example, the security interceptor intercepts and executes security policies, the MTOM interceptor intercepts and executes MTOM policies, and so on. Policies attached to a client or Web Service are executed in a specific order via the Policy Interceptor Pipeline.

The Policy Interceptor Pipeline is shown in the following image:



As shown in the previous image, when a client or a Web Service initiates a message, whether it be a request message in the case of a client or a response message in the case of a Web Service, the policies are intercepted in the following order: Reliable Messaging, Management, Addressing, Security, and MTOM. When a client or a Web Service receives a message, that is, a request message in the case of the Web Service or a response message in the case of a client, the policies are executed in the reverse order: MTOM, Security, Addressing, Management, and Reliable Messaging.

A message may have one or more policies attached. Not every message will contain each type of policy. A message may contain a security policy and an MTOM policy. In this instance, the security interceptor executes the security policy and the MTOM interceptor executes the MTOM policy. In this example, the other interceptors are not involved in processing the message.

The following steps describe how the policy interceptors act on messages between the client and the Web Service:

1. The client sends a request message to a Web Service.
2. The policy interceptors intercept and execute the policies attached to the client. After the client policies are successfully executed, the request message is sent to the Web Service.
3. The request message is intercepted by policy interceptors which then execute any service policies that are attached to the Web Service.
4. After the service policies are successfully executed, the request message is passed to the Web Service. The Web Service executes the request message and returns a response message.
5. The response message is intercepted by the policy interceptors which execute the service policies attached to the Web Service. After the service policies are successfully executed, the response message is sent to the client.
6. The response message is intercepted by the policy interceptors which execute any client policies attached to the client.
7. After the client policies are successfully executed, the response message is passed to the client.

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Explain Web Services and composite security:** We discussed in detail about the Web Service security features
- **Describe the Fusion Middleware security architecture:** We discussed in detail about the Oracle Fusion Middleware security architecture
- **Understand security policy management and identity propagation:** We discussed about configuring security policies and identified the features available to propagate the identity information
- **Describe OWSM Agents:** We discussed in detail about the OWSM Agents architecture



## Self-review questions

1. Oracle WebLogic Server provides support for securing Web Services.
  - a. True
  - b. False
2. Oracle Web Service Manager is a standalone component provided in Oracle Fusion Middleware 11g.
  - a. True
  - b. False
3. Identify the components of Oracle Web Service Manager architecture.
  - a. Oracle WSM Policy Manager
  - b. Oracle WSM Agent
  - c. Metadata Services
  - d. Policy Interceptors
  - e. All the above
4. Identify the types of policies supported by Oracle Fusion Middleware 11g.
  - a. WS-Reliable Messaging
  - b. WS-Addressing
  - c. WS-Security
  - d. MTOM
  - e. All the above
5. WS-Policies can be attached to JDeveloper and Fusion Middleware Console.
  - a. True
  - b. False

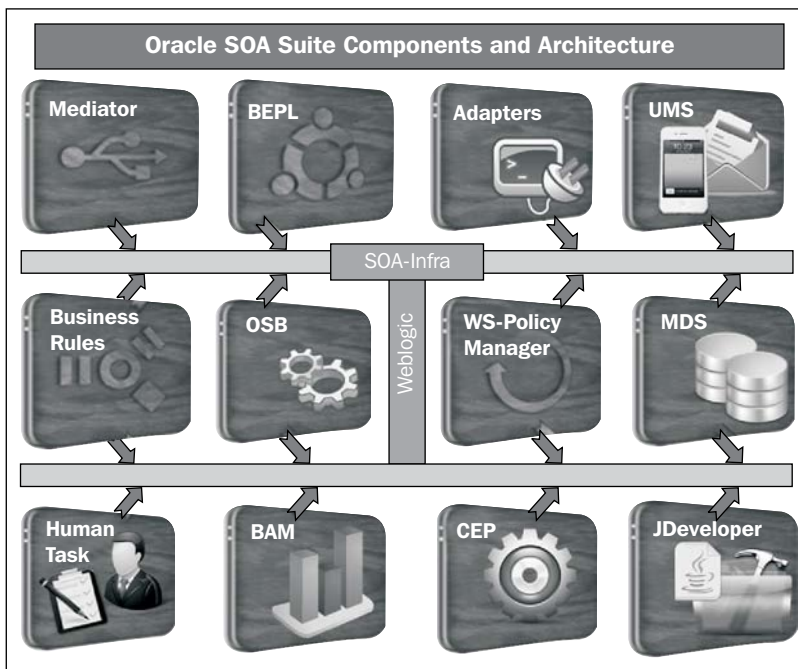
6. Identify the standards that shape Web Service security; choose all the right answers.
  - a. Transport Layer Security – SSL
  - b. XML Encryption
  - c. WS-Security
  - d. WS-Policy
  - e. WS-ReliableMessaging
  - f. WS-SecurityPolicy
  - g. Web Services Addressing
  - h. All the above
7. Identify the statements that correctly identify the architectural changes between Oracle Web Service Manager 10g and 11g. Choose all the correct answers from the following list:
  - a. Oracle WSM Agent functionality is integrated into Oracle WebLogic Server.
  - b. Policy management and monitoring is integrated into Oracle Enterprise Manager Fusion Middleware Control.
  - c. Oracle WSM Policy Manager enforces additional Web Service QoS requirements.
  - d. Oracle WSM Database is replaced by the Oracle Metadata Repository and Oracle Fusion Middleware Database.
  - e. All the above.
8. Identify the default policies defined along in Oracle Fusion Middleware 11g.
  - a. Security policies
  - b. WS-Addressing policies
  - c. MTOM Attachment Policies
  - d. Reliable Messaging Policies
  - e. Management policies
  - f. All the above

9. Your Oracle SOA composite is running in production but due to a new government mandate you have to update the security policy based on Oracle Web Service Manager. What are your options to update the security policy?
  - a. Attach the updated policy via the command-line interface.
  - b. Attach the updated policy in JDeveloper and redeploy the component.
  - c. Attach the policy in Enterprise Model Console, and then test it and reattach without redeployment.
  - d. Attach the policy via the WebLogic admin console after deployment.
10. In Oracle SOA Suite 11g, embedded security is enabled via \_\_\_\_\_.
  - a. Oracle Service Bus as a security gateway.
  - b. Embedded OWSM agents to the SOA service infrastructure.
  - c. Out-of-the-box installation of the OWSM gateway.
  - d. Custom installation of the OWSM gateway.

# 11

## SOA Integration Points

**Oracle SOA Suite** is a complete suite that contains various service engines and adapters that can be readily integrated with other SOA components to enable a true SOA ecosystem in an enterprise. The following diagram shows a view of the various SOA/SCA components that can be integrated together to enable a true SOA ecosystem:



This chapter will cover the following list of exam objectives:

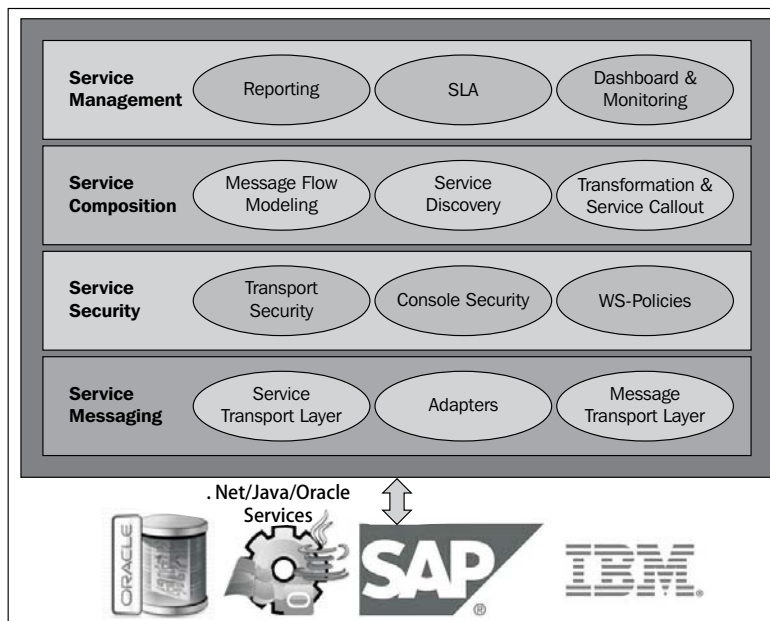
- The Oracle Service Bus concepts and architecture
- Oracle Business Activity Monitoring
- Oracle B2B

## The Oracle Service Bus concepts and architecture

**Service bus** is an ideal architecture for integrating applications through services. It uses the applications to expose the data through services and implements the service orchestration or application integration using **Oracle Service Bus (OSB)**. The features provided by the service bus are to filter, transform, validate, and enrich data to complete the complex application integration.

### OSB product architecture

The OSB product consists of components that help to accomplish virtualization of enterprise system resources to expose them as services: Service Management, Service Composition, Service Security, and Service Messaging features required for implementing SOA. The following diagram explains the OSB product architecture:



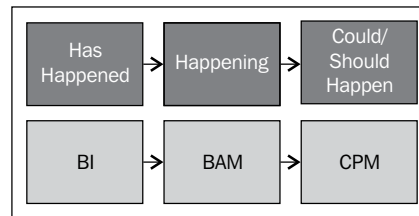
Key concepts to be understood about OSB are stated as follows:

- **Location transparency:** Oracle Service Bus creates location transparency through isolation of consumers from service providers and acts as a proxy layer to transport the message between the service consumer and provider. This feature is addressed by OSB as **service virtualization**.
- **Dynamic routing:** Oracle Service Bus provides features to dynamically route the messages through **message pipelines**.
- **Message enrichment:** Oracle Service Bus provides the required features to enrich the messages through service callouts and message transformation activities.
- **Service orchestration:** Oracle Service Bus can interact with multiple services to expose the data to orchestrate the business process or create a new service by combining different services.
- **Service pooling:** OSB supports load balancing of service endpoints through service pooling technique.
- **Data transformation:** Data is transformed from one form to another using XSL, XQuery, or MFL.
- **Protocol transformation (adapters):** Protocol transformation can be achieved using adapters. OSB supports the following messaging protocols:
  - HTTP/HTTPS
  - SOAP
  - JMS
  - FTP
  - File
  - DB
  - SMTP
  - Tuxedo
  - JDBC
  - MQ
- **OSB patterns:**
  - **VETRO pattern:** Validate-Enrich-Transform-Route-Operate are the patterns used by OSB to enable service virtualization.

- **Split-join:** OSB provides the feature to split the incoming request to process through multiple parallel pipelines and join the results to increase the throughput and performance of the service response. This pattern is applicable for use cases in which the services are not dependent on each other to produce the desired result for a given moment of interaction.

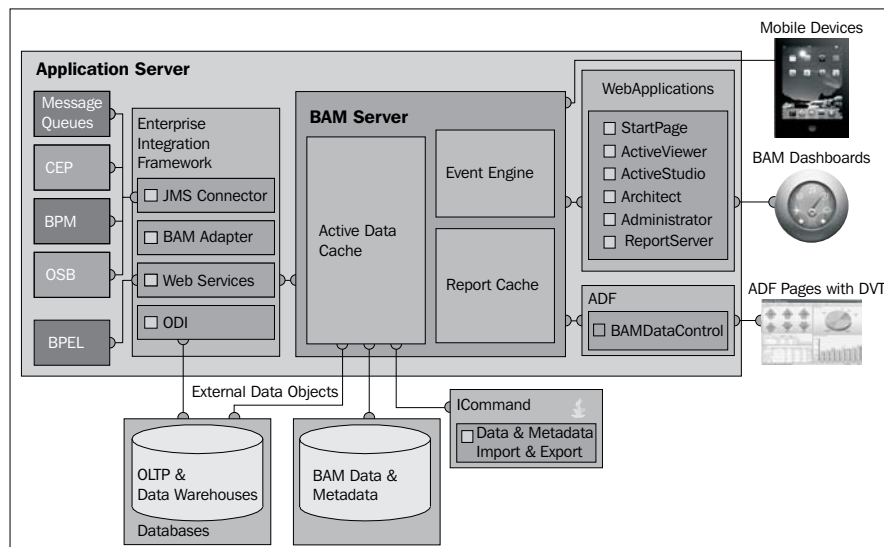
## Oracle Business Activity Monitoring

**Oracle Business Activity Monitoring (BAM)** enables us to monitor business services and processes in an enterprise, correlate **Key Performance Indicators (KPIs)**, and change business processes or take corrective action if the business environment changes.



## The Oracle BAM architecture

The following diagram shows the Oracle BAM architecture:



## The Oracle BAM server

The Oracle BAM server contains the following list of components:

- **Active Data Cache:** Data Cache holds the messages and information from sensors, adapters, and other EIS layers to serve the BAM reports
- **Event Engine:** Event Engine helps to process the alert messages and notifications to the users
- **Report Cache:** Report Cache holds the data objects used in reporting change list and snapshots of the report details

## The Oracle BAM web applications

The Oracle BAM web applications include Active Viewer, Active Studio, Architect, and Administrator, which are described as follows:

- **Active Viewer:** Active Viewer is the thin user interface for the business user. When new information is available, the user receives an instant message that contains a link to the information.
- **Active Studio:** Active Studio is the thin user interface for the power user. Using Active Studio, the power user can create and edit reports. Reports can be shared with other users and rules can be created for determining the scheduling and delivery of the reports. Report types include columnar reports, crosstabs, KPIs, charts, spreadsheets, and more.
- **Architect:** Architect is the thin user interface for the data designer. Using Architect, the data designer creates and manages data objects in the Active Data Cache and manages real-time message processing.
- **Administrator:** Administrator is the thin user interface for the system administrator who is responsible for user management and overall server management.

## The Oracle BAM ADF control

The Oracle BAM ADF control is used to integrate BAM data to the ADF application to build custom actions on the BAM reports.

## Integrating BAM with Oracle SOA Suite

BAM can be integrated with Oracle SOA Suite using BAM Adapters or BAM Sensors.



## Oracle B2B

Oracle B2B is a gateway that enables secure and reliable exchange of business documents between an enterprise and its trading partners. Oracle B2B supports business-to-business document standards, security, transports, messaging services, and trading partner management. Oracle B2B is a binding component that enables the implementation of e-commerce business processes. Oracle B2B supports transformation and transfer of EDI messages between different trading partners.

## Features of an Oracle B2B product

The features of the Oracle B2B product are stated as follows:

- **Trading partner management:**
  - Ability to create trading partners and associates required to document and exchange protocols
  - Ability to clone trading partners
- **Document management:**
  - Supports multiple document types
  - Ability to create custom document definitions and create extensions to various other standard documents through the document editor
- **Agreement management:**
  - Ability to create, edit, deploy, retire, and purge agreements
  - Monitoring and metric reports
  - Ability to monitor transactions and generate metric reports
- **Integration:**
  - Extensive integration support with Oracle E-Business and Oracle SOA Suite
- **Certificate management:**
  - Enterprise **Java Key Store (JKS)** is used to store common keys and are used across the enterprise

- **Enhanced features of Oracle B2B 11g:**
  - Rich and enhanced user interface.
  - Easier wizard-based trading partner setup.
  - Ability to clone the trading partner details.
  - UI-based resubmission of inbound and outbound messages.
  - Centralized user management in the WebLogic Server.
  - Ability to create EDI batches from the Oracle B2B application. Complex batches can be configured from B2B UI.
  - Metric reports are available in B2B 11g.

## Industry-standard protocols supported by Oracle B2B

The following table shows the Industry standard protocols supported by Oracle B2B.

Topic	Oracle B2B
Industry standards	EDI, UCCNet, RosettaNet, CIDX, ebXML, PIDX, UBL, and positional flat files.  Covers all the industry standards.
Document protocols	X12, EDIFACT, HL7, OAG, cXML, custom document definition, xCBL, NCDCP, UBL, and EDIEL.  Covers all the industry-standard document protocols.
Transport protocols and exchanges	File, FTP, SFTP, HTTP, HTTPS, SMTP, TCP/IP, AQ, JMS, AS1, AS2, ebMS, RNIF, and MLLP.  Covers most of the industry-standard transport protocols and exchange patterns.
Message formats	MIME, SMIME, XML, XMLDSig, XMLEncrypt, and SOAP.  Covers all the industry-standard message formats for B2B communication.

## Oracle B2B 11g home page

The following screenshot shows the home page of the Oracle B2B application with a list of trading partners and their agreements:

The screenshot displays the Oracle B2B 11g application interface. The top navigation bar includes links for Administration, Partners, Reports, Metrics, Help, and Logout. The user is logged in as 'GChips'. The main content area is divided into two panes. The left pane shows a list of trading partners, with 'Acme' selected. The right pane displays the 'Acme' trading partner profile, which includes sections for Identifiers, Contact Information, Parameters, and Key Store.

**Oracle B2B** Administration Partners Reports Metrics Help Logout  
Logged in as GChips

**Partner** Search Name Advanced  
Acme  
GChips

**Agreement** Search Name Advanced  
Acme\_GChips\_X12\_4010\_850\_File

**Profile** Users Documents Channels

**Acme** Save Export  
The trading partner profile uniquely identifies each partner. Set up identifiers, contact information, and customized parameters for each partner.

**Identifiers** + -  
Identifier types uniquely identify a trading partner and define how to exchange documents.

Type	Value
Name	Acme
EDI Interchange ID	Acme
EDI Group ID	Acme
EDI Interchange ID Qualifier	ZZ

**Contact Information** + -  
Important contact information for each trading partner should be entered.

Type	Value
Phone	13105551212

**Parameters**  
Additional customized parameters can be created and assigned to each trading partner.  
No parameters exist for this section

**Key Store**  
Password Location  
Confirm Password

Copyright © 2004, 2009, Oracle and/or its affiliates. All rights reserved.

## Oracle B2B deploy agreement

The following screenshot shows the list of deploy agreements in the Oracle B2B gateway:

**Oracle B2B** Administration Partners Reports Metrics Help Logout  
Logged in as

Document Deploy Manage Deployments Types Import/Export Schedule Batch Manage Batch Callout Purge Listening Channel Configuration

**Deploy Agreement**  
Deploying an agreement is the process of activating an agreement from the design-time repository to the run-time repository.

Search Advanced Saved Search Default

Match ☐ All ☒ Any

Name   Document Protocol Name

Trading Partner   Document Protocol Version

Identification Value   Document Type

Identification Type   Document Definition

**Agreement**

Agreement	Initiating Partner	Responding Partner	Supported Document	Channel	State	Last Deployed
				From To		
Acme_GChips_X12_4010_850_File	Acme	GChips	ED1_X12 - 4010 - 850 - 850def	GChips_File_Endpt	Draft	
GChips_ED1_X12_4010_850_850def_Inbound	GChips	Acme	ED1_X12 - 4010 - 850 - 850def	GChips_File_Endpt	Draft	
GChips_ED1_X12_4010_997_997def_Outbound	Acme	GChips	ED1_X12 - 4010 - 997 - 997def	GChips_File_Endpt	Draft	

## Oracle B2B metrics report

The following screenshot shows the Oracle B2B metrics details:

**System** Partners

**Summary** Chart Timeline in Hours: 10 Refresh

**Summary**  
Number of Active Partners: 3  
Number of Active Agreements: 14  
Number of Active Document Types: 5

**Messages and Errors**

**Message Count**

**Active Document Types**

Name	No. Of Messages Processed	Average Processing Time (msec)	Average Message Size (kb)	Errors
	Outbound	Inbound	Outbound	Inbound
EDI_ED1FACT-998A-ORDERS	2	12	5,266	5,065.08
HL7v2.3.1-ADT_A01	2	2	3,108.5	3,599
EDI_X12-4010-850	2	12	9,786.5	9,938.67
Custom-1.0-ORDERS_FILE	54	56	230.54	650.79
UserDefined-1.0-ORDERS_FTP	2	2	173.5	176.5

**Active Trading Partners**

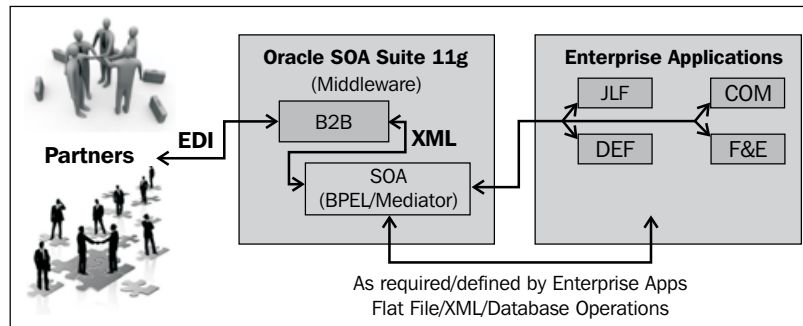
Name	No. Of Messages Processed	Average Processing Time (msec)	Average Message Size (kb)	Errors
	From	To	From	To
Acme	62	94	792.23	2,397.61
GlobalChips	17	10	4,235.41	3,735.7
GlobalParts	70	52	2,174.79	226.17

**Errors**

Error Code	Error Text	Initiating Partner	Responding Partner	Document Type	Timestamp	Business Message Id
B2B-50547	Agreement n...	Acme	Acme	CONFIRM_BOD	2009-04-30 13:18...	8C5784CD120F8ADE2C0000000B75000
B2B-50547	Agreement n...	Acme	Acme	CONFIRM_BOD	2009-04-30 13:19...	8C5784CD120F8AEB17D0000000B82000
B2B-50547	Agreement n...	GlobalChips	Acme	PROCESS_PO	2009-04-30 13:19...	8C5784CD120F8AEB1D0000000B89000
B2B-50547	Agreement n...	Acme	Acme	PROCESS_PO	2009-04-30 13:19...	8C5784CD120F8AEB2630000000B90000

## Oracle B2B Reference Integration Architecture

Oracle B2B can be integrated with Oracle SOA Suite using a B2B Adapter or JMS/AQ Adapters, based on the queue configuration done in the B2B application. Oracle B2B provides `IP_IN_QUEUE` and `IP_OUT_QUEUE` to process inbound and outbound messages. Errors generated from the Oracle B2B gateway are queued in `IP_IN_QUEUE` with the consumer name as `b2berroruser`.



## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Describe the Oracle Service Bus (OSB) concepts and architecture:** We discussed in detail about the OSB concepts to accomplish virtualization of services
- **Describe Oracle Business Activity Monitoring (BAM):** We discussed in detail about BAM and its significance
- **Describe Oracle B2B:** We identified the need for B2B and features supported by it to enable trading partner integration

## Self-review questions

1. OSB complements the SOA stack for which of the following specified purpose.
  - a. Service virtualization
  - b. Platform virtualization
  - c. Application virtualization
  - d. Service registration
2. Which of the following list of service transport protocols are supported by OSB. Choose all the correct answers.
  - a. EJB/RMI
  - b. CORBA
  - c. IIOP
  - d. JMS
3. Mediator can be used as a substitute for OSB.
  - a. True
  - b. False
4. JDeveloper can be used to develop OSB components.
  - a. True
  - b. False
5. Identify the integration patterns supported by OSB. Choose all the right answers from the following list:
  - a. Fan-out
  - b. Split-join
  - c. Dynamic routing
  - d. Master details
6. Identify the integration patterns supported by B2B. Choose all the right answers from the following list:
  - a. Fan-out
  - b. Split-join
  - c. Dynamic routing
  - d. Master details

7. OSB supports transformation of data using:
  - a. XQuery
  - b. XSL
  - c. MFL
  - d. All of the above
8. EDI messages can be handled using:
  - a. Oracle B2B
  - b. Oracle BAM
  - c. Oracle CEP
  - d. None of the above
9. Oracle B2B 11g can be integrated with Oracle SOA Suite SCA components using which of the following? Choose all the right answers.
  - a. B2B Adapter
  - b. JMS Adapter
  - c. AQ Adapter
  - d. File Adapter
10. Oracle B2B 11g provides tools to automate the agreement deployments to different environments.
  - a. True
  - b. False
11. Oracle B2B 11g mandates the creation of unique document definitions for each agreement.
  - a. True
  - b. False
12. Oracle B2B 11g does not provide an option to resubmit the messages.
  - a. True
  - b. False
13. Oracle B2B 11g provides an option to auto-create agreements.
  - a. True
  - b. False

14. Oracle B2B 11g provides an option to store partner contact information.
  - a. True
  - b. False
15. Oracle B2B is a part of Oracle EDN Suite enabling web service mediation.
  - a. True
  - b. False
16. Oracle B2B is registered as a service engine in the Oracle SOA infrastructure.
  - a. True
  - b. False
17. Oracle B2B provides the feature to auto-generate functional acknowledgements.
  - a. True
  - b. False
18. Inbound Oracle B2B errors cannot be handled.
  - a. True
  - b. False
19. Identify the default queues provided by the Oracle B2B product from the following list:
  - a. IP\_IN\_QUEUE
  - b. IP\_OUT\_QUEUE
  - c. B2B\_IN\_QUEUE
  - d. B2B\_OUT\_QUEUE
  - e. B2B\_ERROR\_QUEUE
20. Default queues provided by Oracle B2B are Oracle AQs.
  - a. True
  - b. False
21. Oracle B2B provides the feature to batch the messages.
  - a. True
  - b. False



22. Oracle B2B provides the feature to call Java programs and XSLTs through the callout feature.
- a. True
  - b. False
23. Oracle B2B provides inbuilt features to encrypt and decrypt the data.
- a. True
  - b. False
24. Oracle B2B supports ebMS 2.0 specifications for error handling.
- a. True
  - b. False
25. Complex Event Processing is supported through \_\_\_\_\_ and \_\_\_\_\_.  
Choose the right answers from the following list:
- a. CQL
  - b. EPL
  - c. BPEL
  - d. EDL
26. The event definition for Oracle SOA Suite EDN is supported through:
- a. EDN
  - b. BPEL
  - c. EDL
  - d. OSB
27. Identify the list of Oracle SOA Suite components that can publish and subscribe events. Choose the right answers from the following list:
- a. Mediator
  - b. BPEL
  - c. Human Task
  - d. Business Rules

28. Fault handling is not supported by the EDN layer in Oracle SOA Suite.
- a. True
  - b. False
29. Oracle CEP programs can be developed using JDeveloper.
- a. True
  - b. False
30. Oracle CEP Visualizer is preinstalled along with the CEP server.
- a. True
  - b. False
31. Identify the type of BPEL Sensors. Choose the right answers from the following list:
- a. Activity
  - b. Adapter
  - c. Variable
  - d. Fault
  - e. Service
32. Choose all the components that are part of the BAM Servers.
- a. Oracle BAM Process Manager
  - b. BAM Active Data Cache
  - c. BAM Event Engine
  - d. BAM Report Cache
  - e. BAM Sensor Network
33. BAM Server can be integrated with Oracle SOA Suite through which of the following options? Choose all the right answers.
- a. Oracle BAM Adapter
  - b. Oracle BAM Sensors
  - c. JMS Adapter
  - d. DB Adapter
  - e. File Adapter

34. Which of the following Oracle BAM components allow ADF developers to build applications with a dynamic user interface that changes based on real-time business events.
- a. BAM Administrator
  - b. BAM Report Server
  - c. BAM Architect
  - d. BAM Data Control
35. Oracle BAM objects can be migrated using the ICommand utility.
- a. True
  - b. False

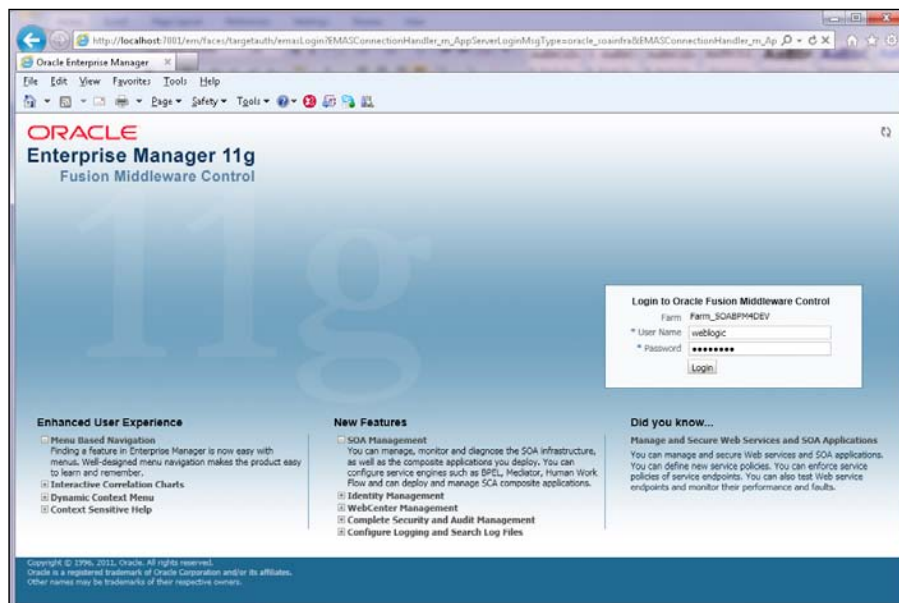
# 12

## Monitoring and Managing SOA 11g Deployment

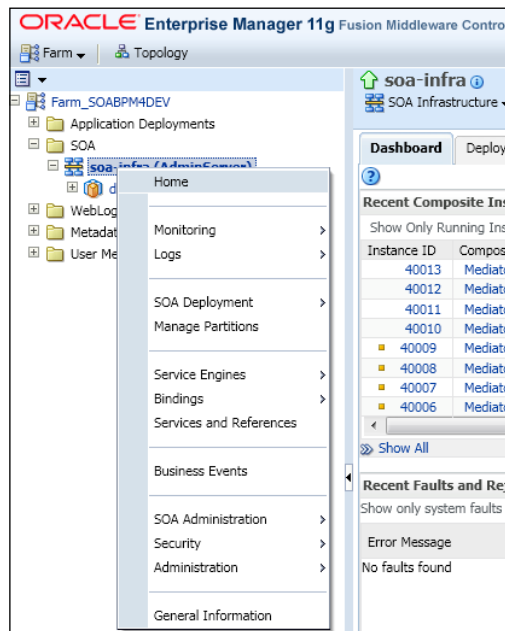
Monitoring and managing SOA 11g can be done from **Oracle Enterprise Manager 11g Fusion Middleware Control** and **SOA Management Pack for Oracle Enterprise Manager 11g**.

Oracle Enterprise Manager can be launched in your environment using the following link format in the case of the default Oracle SOA Suite installation procedure. If custom installation procedures are followed in your environment, contact your system administrators for more details.

`http://<machine _name>:<port_number>/em` (default port number is 7001).



After you login to the **Enterprise Manager (EM)** console for SOA, you will be able to manage SOA composites and view the **SOA Infrastructure Dashboard**. **SOA Dashboard** will provide details about the SCA instances that were executed recently, **Deployed Composites**, adapter details, SCA components metric details, and a performance summary of various **Service Engines** and binding components details.



This chapter will cover the following exam objectives:

- Describing, deploying, and undeploying the SOA composite application: Readers of this section will be able to understand the deploy options that are available in the SOA composite application.
- Describing management of the SOA composite application using the Enterprise Manager
- Describing the Enterprise Manager Grid Control in the SOA Management Packs

## Deploying and undeploying the SOA composite application

Deploying the application to SOA runtime can be performed in the following three ways:

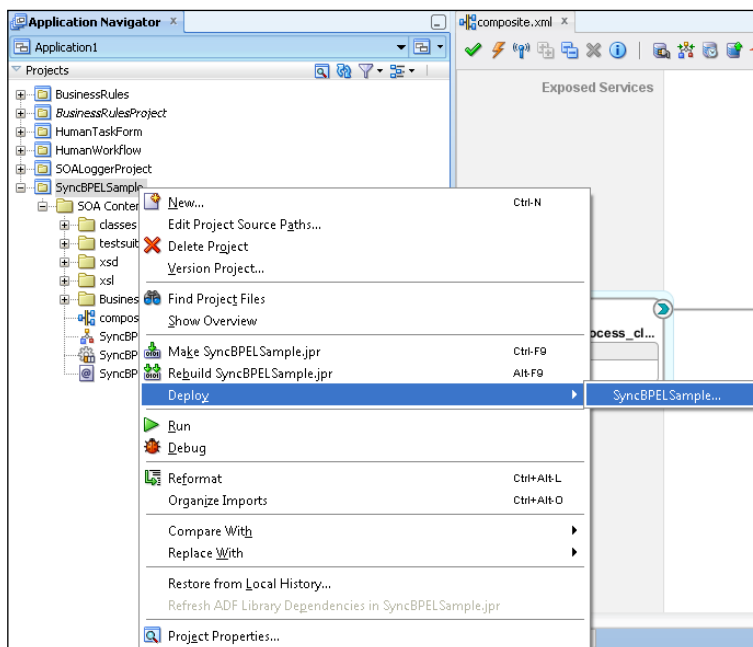
- Using JDeveloper to deploy the composites.
- Using the SOA EM SOA deployment wizard to deploy the SAR file generated through JDeveloper.
- Using custom Ant-based tasks and WSLT functions provided by Oracle. This method is used only by administrators and deployment managers in cases of large SOA-scale programs. In other cases, developers are restricted with the previous two options.

We will show, in detail, each of the previously discussed options.

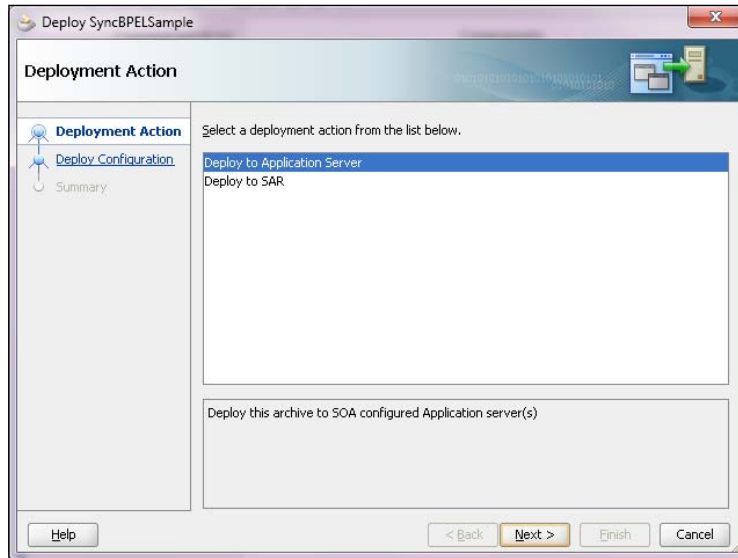
### Using JDeveloper to deploy the composites

To deploy the composites using JDeveloper, we need to perform the following steps:

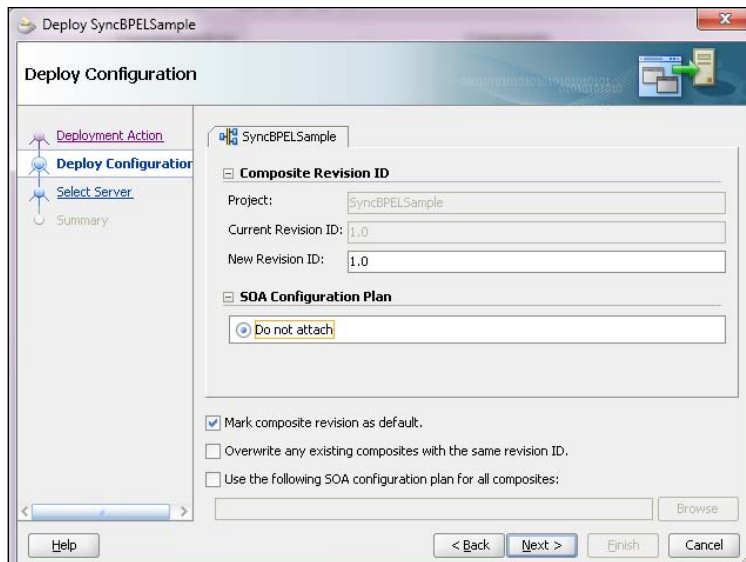
1. Choose the **Deploy** option from JDeveloper by right-clicking on the project tree node in the SOA application.



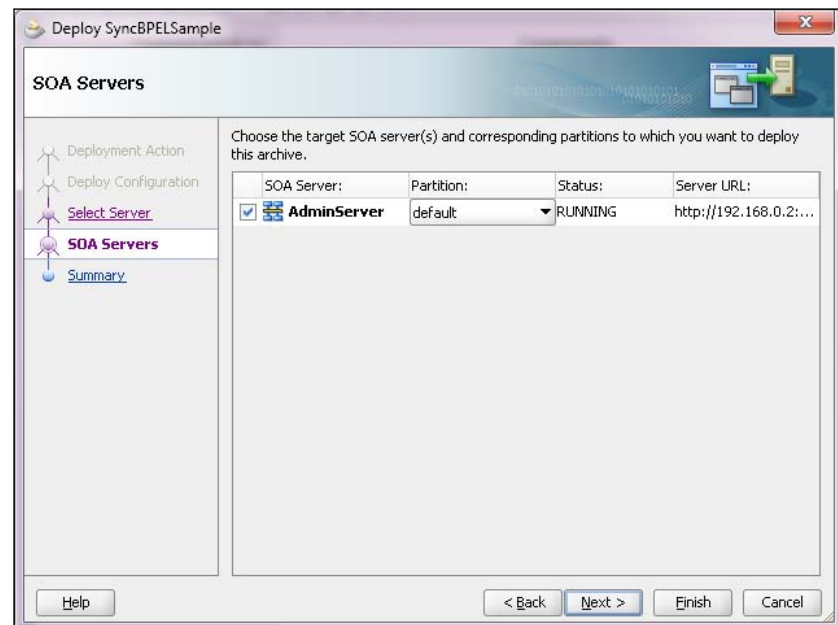
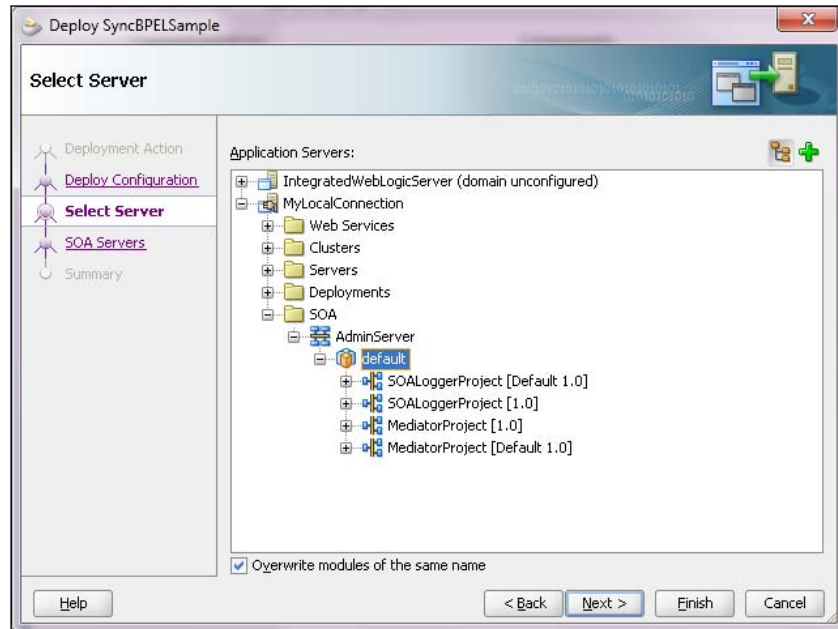
2. Choose the **Deploy to Application Server** to directly deploy the composite to the SOA EM console.



3. We are now provided with an option to attach the deployment configuration plan and revision ID. We will discuss configuration plan in detail later in this chapter. Revision ID helps to specify the version of the composite being deployed to the console.

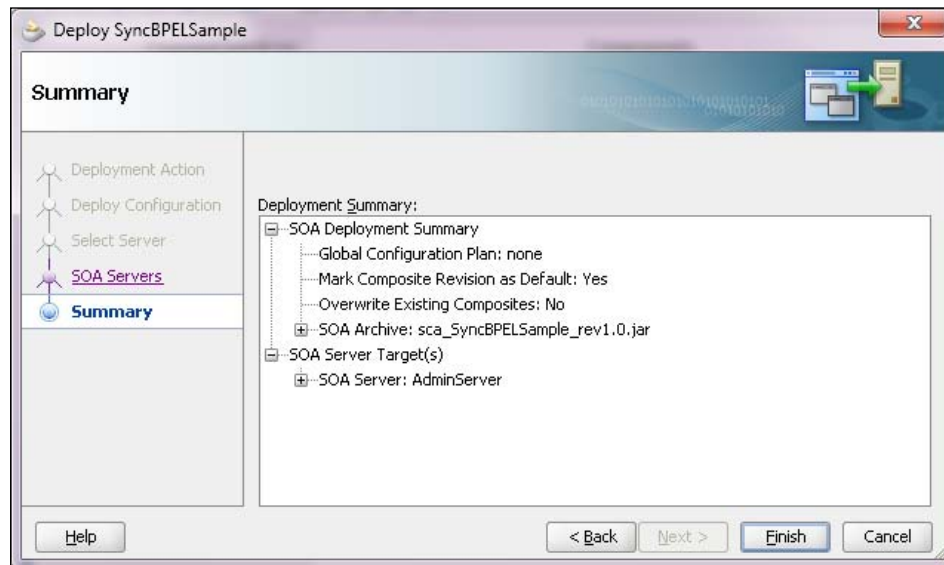


4. Select the server and the partition in which the composite application should be deployed. Make the application server connection if this is not available already.

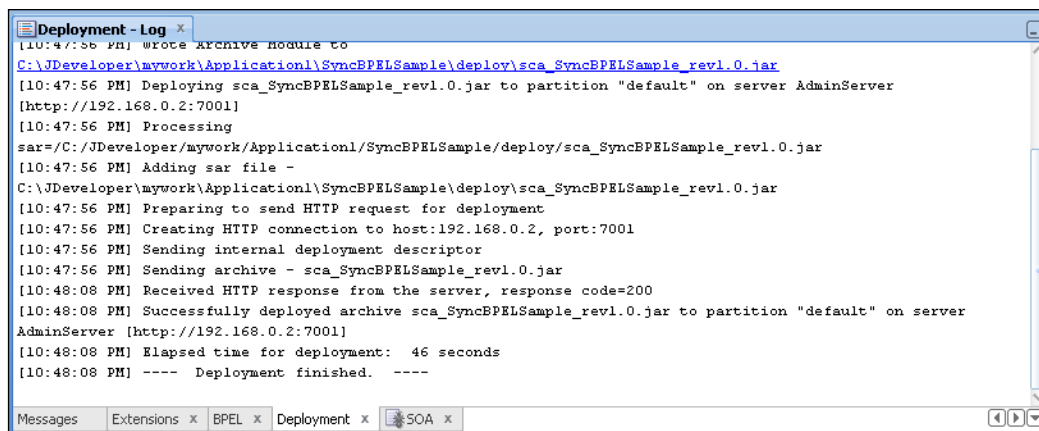




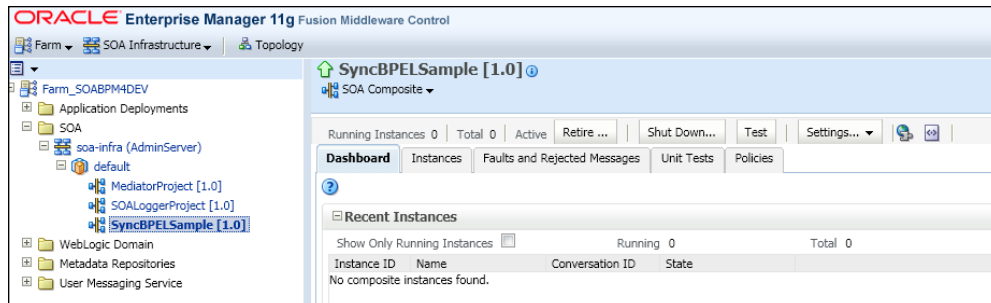
5. **Deployment Summary** can be obtained before starting the deployment. If the details provided are incorrect, the **Back** button can be used to reconfigure the deployment setup.



On clicking the **Finish** button, the deployment will start by running the dependency analysis and ensure that the compilation is successful. On successful deployment, the message shown in the following screenshot will be received; else the error returned by the server during the deployment will be shown in this pane:



The deployed composite can be seen from the console, as shown in the following screenshot:



## Using the SOA EM SOA deployment wizard to deploy the SAR file generated through JDeveloper

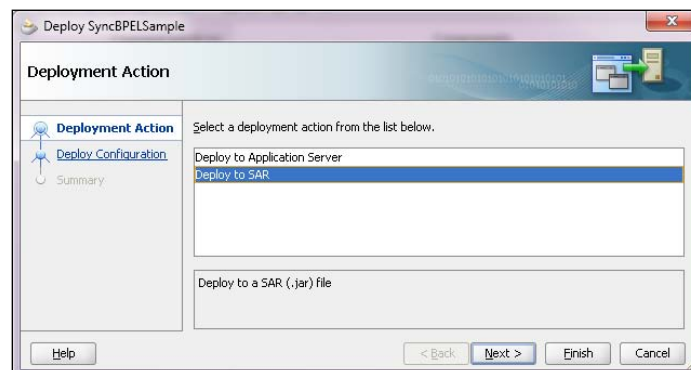
Deploying the composite through the SOA EM console is a two-stage process. This option is quite handy when deploying large composites that take an enormous amount of time to compile and deploy due to the complexity of the composite. The following list shows the stages in the process:

1. Generating the SAR file through JDeveloper.
2. Deploying the SAR file from the console.

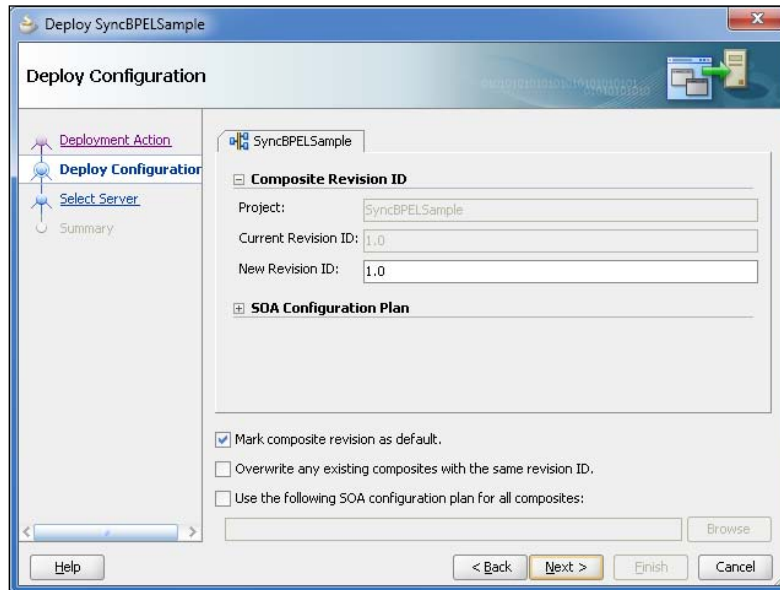
## Generating the SAR file through JDeveloper

The following steps are performed to generate the SAR file through JDeveloper:

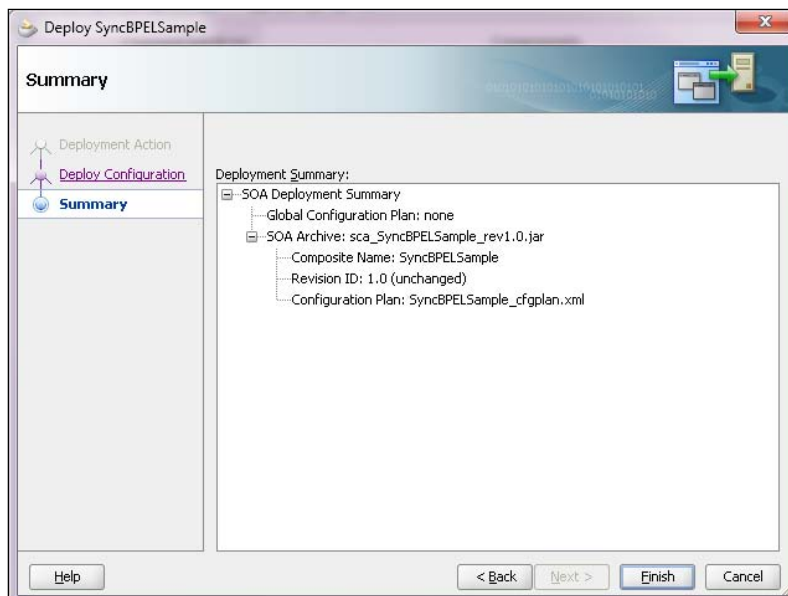
1. Choose **Deploy to SAR** from JDeveloper to use this option.



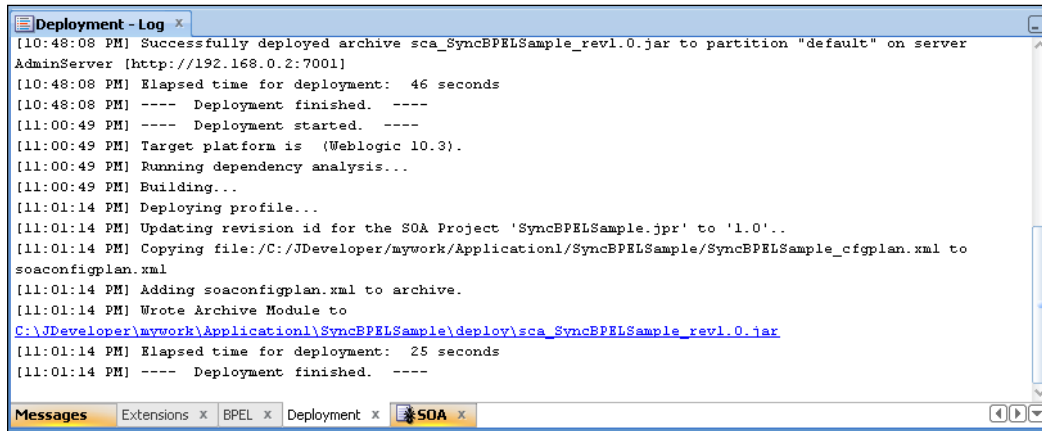
2. Revision the composite and choose the other required applicable configurations to override the composite or attach the deployment configuration, whichever is applicable in your case.



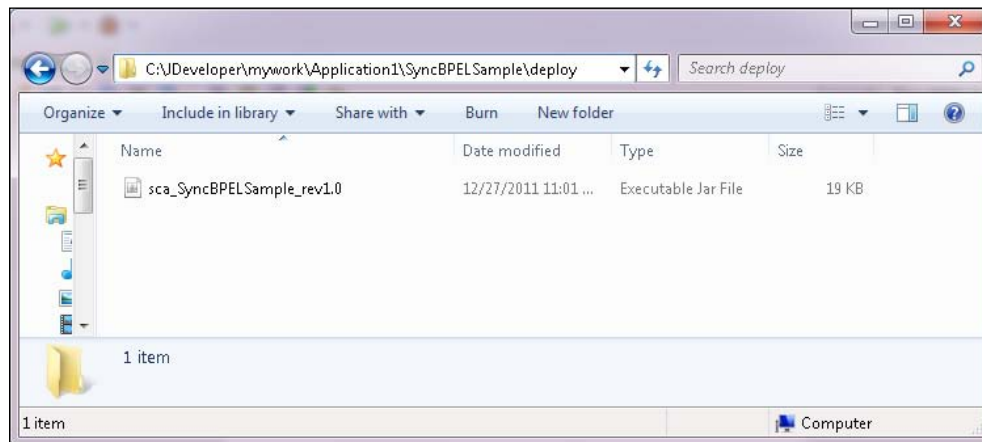
3. Select **Finish**. The wizard will now deploy the SAR file to your filesystem.



The **Deployment-Log** will show the SAR file path in which the composite is deployed, as shown in the following screenshot:



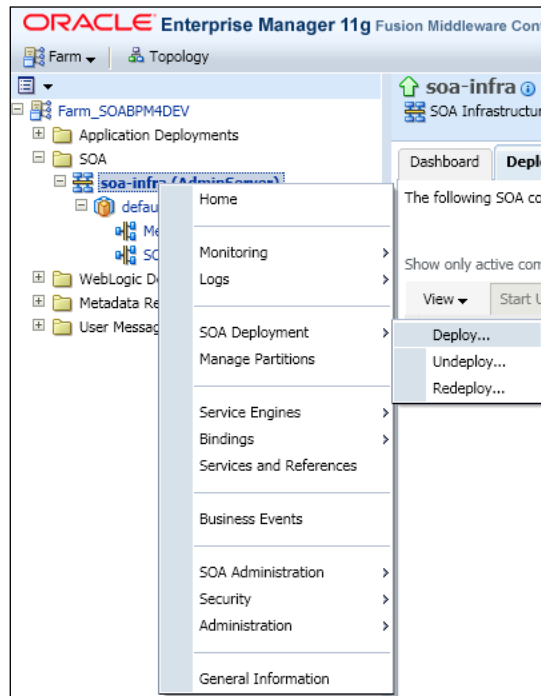
Review the composite version from the filesystem.



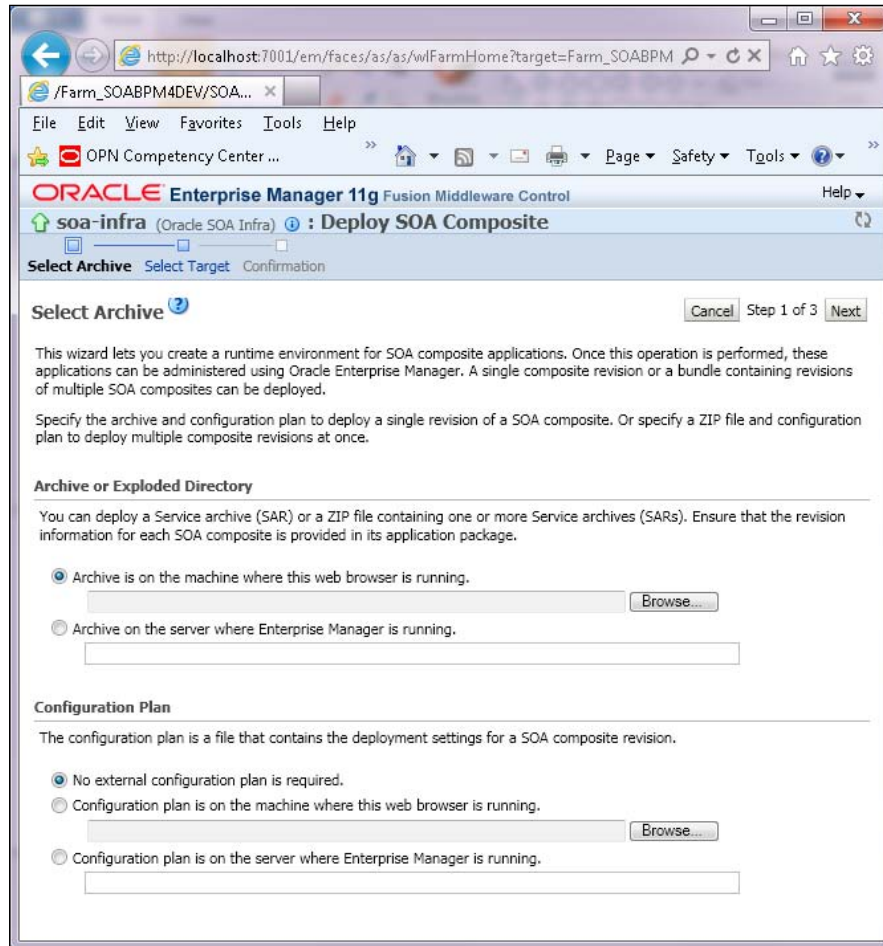
## Deploying the SAR file from the console

The following steps are performed to deploy the SAR file from the console:

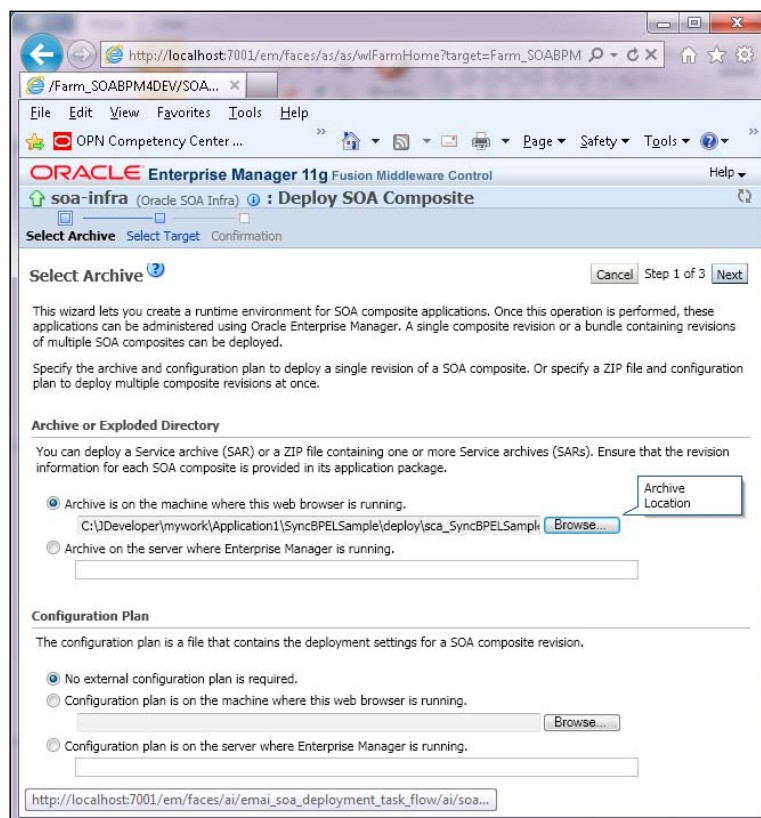
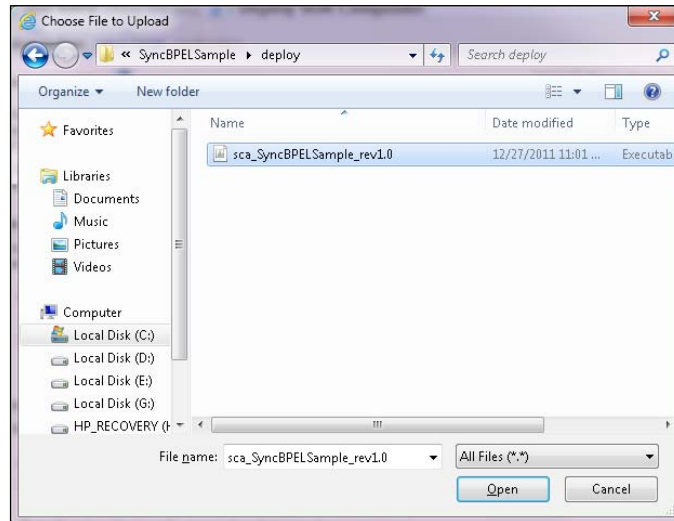
1. Launch the deployment wizard from **Oracle Enterprise Manager 11g**. This can be launched using: `http://<localhost>:<default_port_number>/em`.



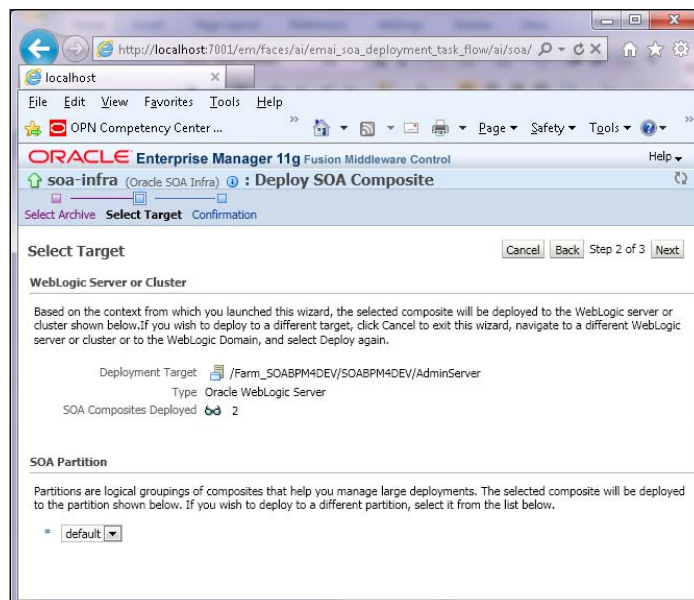
2. Choose the SAR archive to deploy the composite from the **Deploy SOA Composite** wizard, as shown in the next screenshot:



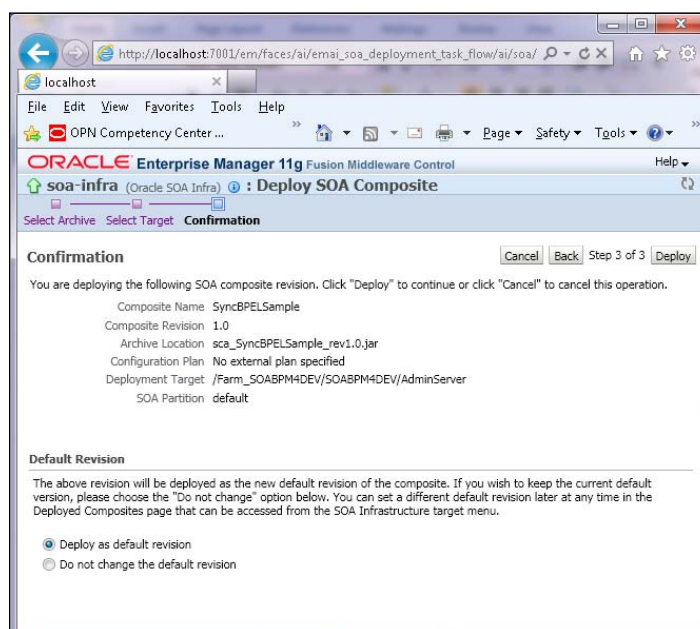
Choose the SAR file, as shown in the next screenshots:



3. Choose the target partition in which this needs to be deployed.

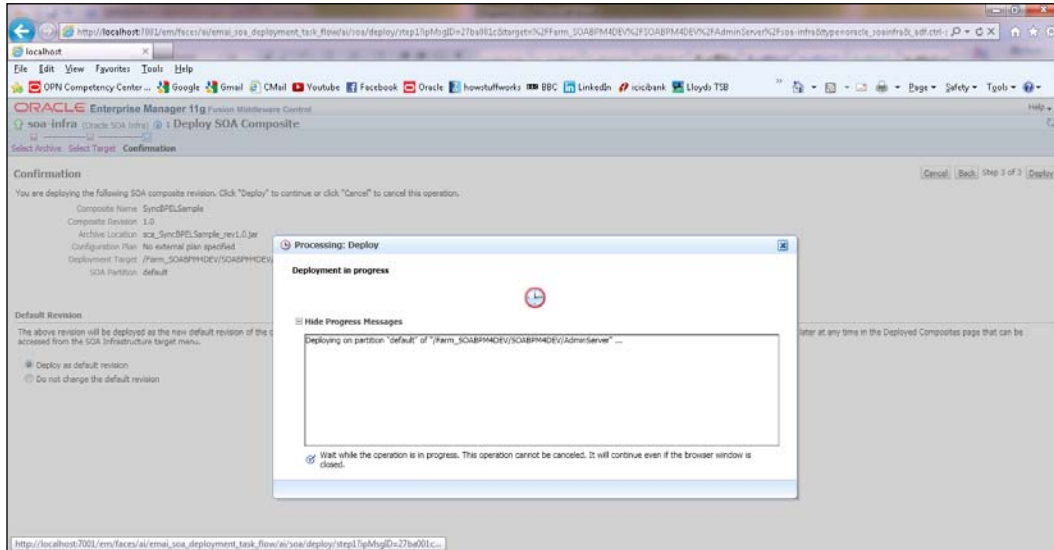


4. The SOA deployment wizard provides an option to override the default version.

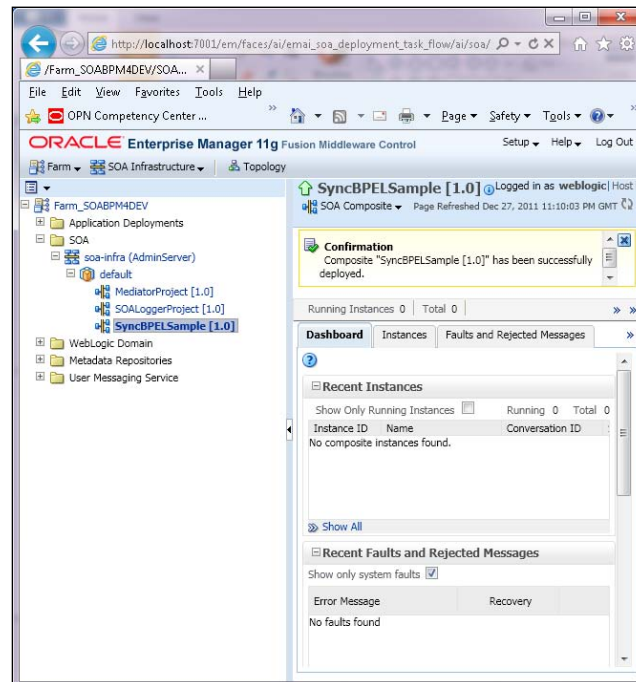




5. The **Finish** button will start the SAR deployment into the SOA EM console.



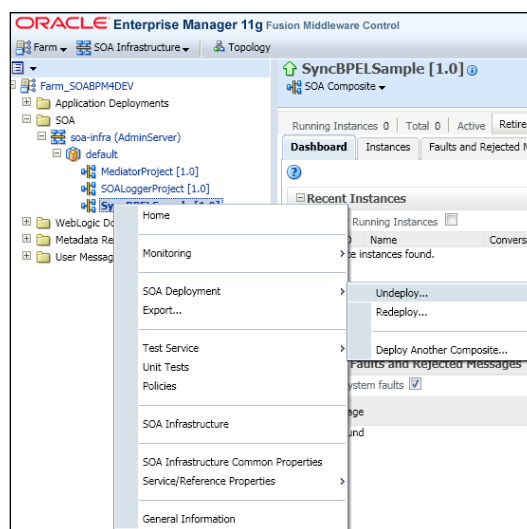
The deployed composite status can be viewed from the SOA console, as shown in the following screenshot:



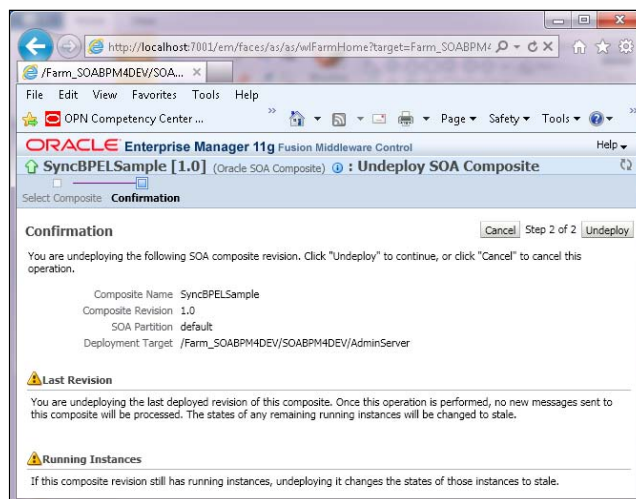
## Undeploying composites

If you are planning to redeploy a composite or override the composite, it is recommended to undeploy the composite. The following steps are performed to undeploy a composite:

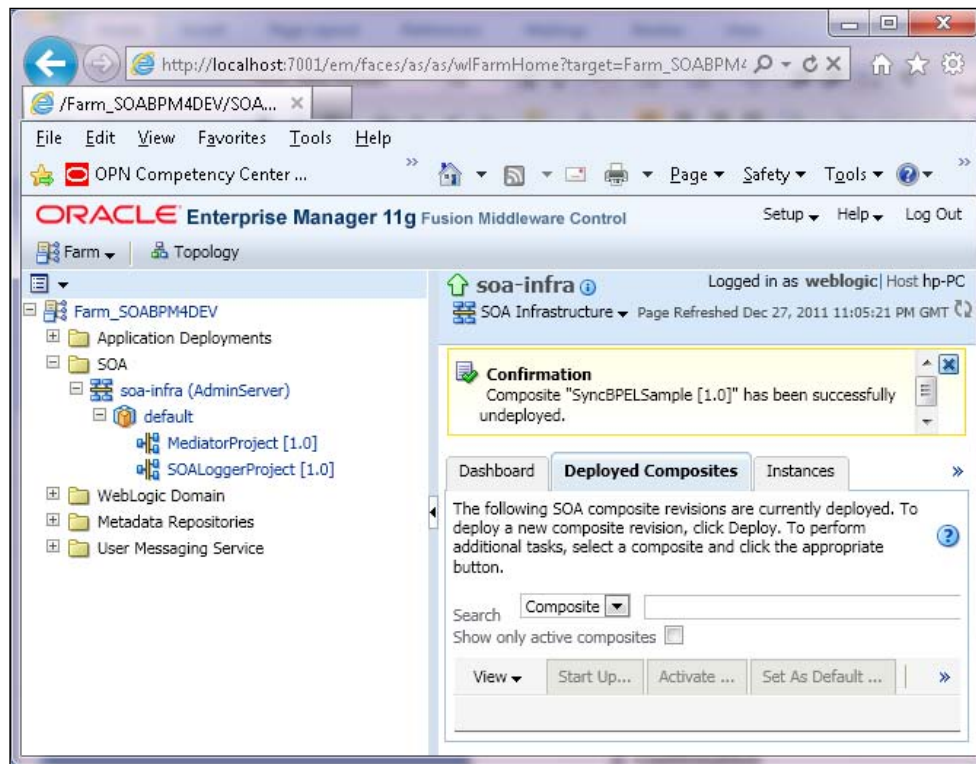
1. Choose the composite to be undeployed. Right-click and choose the **SOA Deployment** option and select the **Undeploy** item to undeploy the composite, as shown in the following screenshot:



2. Confirm the undeployment by clicking on the **Undeploy** button.



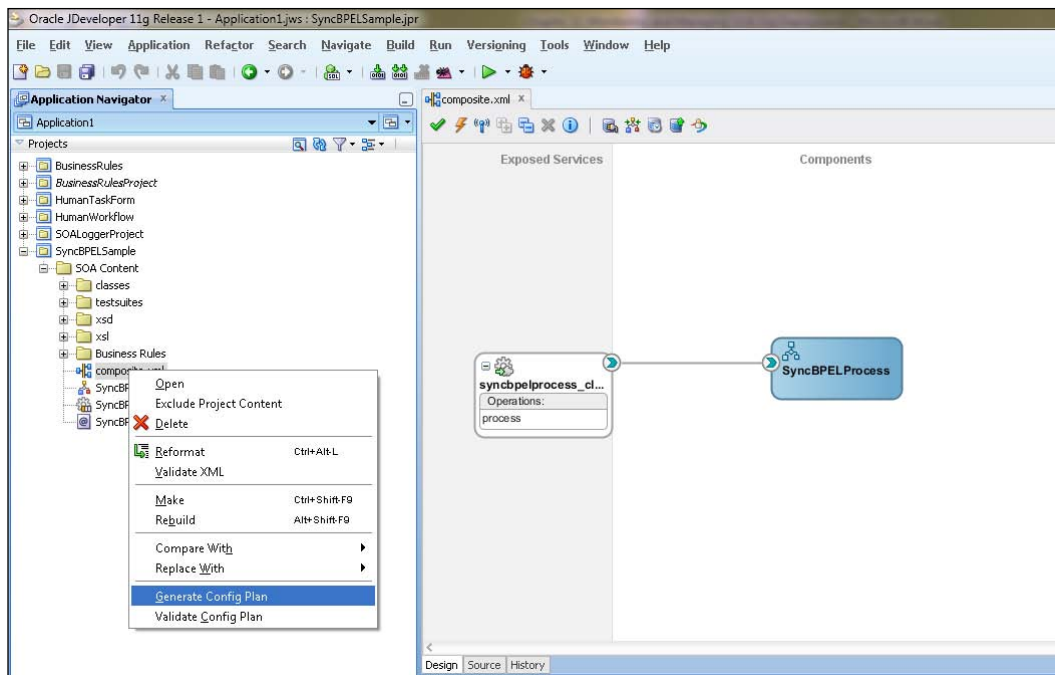
The undeployed composite status can be viewed from the SOA console, as shown in the following screenshot:



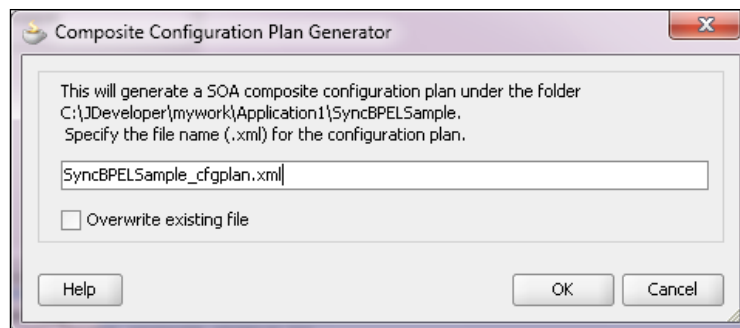
## SOA configuration plan

Understanding the SOA configuration plan will aid the reader to plan the SOA configuration with ease. This will be quite a handy tool to customize the SOA configurations and customize the external reference components using the deployment configuration plan. It uses a simple search and replace framework that searches for the patterns and replaces them with text contents specified in the deployment configuration plan attached with the composite. The configuration plan for an SCA composite can be created using the following specific steps:

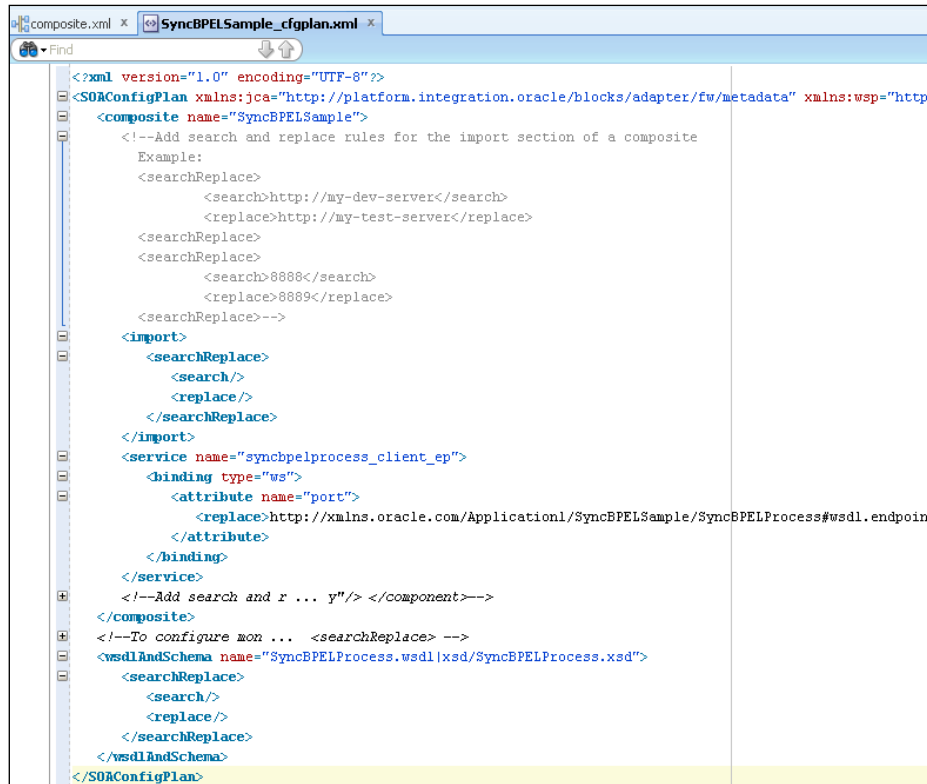
1. The SCA configuration plan template for a composite can be generated using the menu option available by right-clicking on the `composite.xml` file, as shown in the following screenshot:



- Specify the name of the configuration plan. Check with your architecture team and deployment administrator to find out if they are looking for a specific name to be provided for the configuration plan to facilitate automation of composites.



- On clicking **OK** from the previous dialog box, the configuration plan template for the composite will be generated, as shown in the following screenshot:



The following snippet of code shows the sample deployment configuration plan generated from the `composite.xml` file. The configuration plan contains the required details to search and replace tokens that appear in your SCA configuration files. The configuration plan can be attached during the deployment of a composite, which will take care of customizing the references and tokens used in the SCA composite's configuration. Ant build tokens can also be used in this and values for the tokens can be provided in a custom property file. Based on the deployment policies followed in your environment, the usage of the configuration plan will vary.

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAConfigPlan
xmlns:jca="http://platform.integration.oracle/blocks/adapter/fw/me
tadata"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:orawsp="http://schemas.oracle.com/ws/2006/01/policy"
xmlns:edl="http://schemas.oracle.com/events/edl">
```

---

```

xmlns="http://schemas.oracle.com/soa/configplan">
<composite name="SyncBPELSample">
<!--Add search and replace rules for the import section of a
composite
    Example:
<searchReplace>
<search>http://my-dev-server</search>
<replace>http://my-test-server</replace>
<searchReplace>
<searchReplace>
<search>8888</search>
<replace>8889</replace>
<searchReplace>-->
<import>
<searchReplace>
<search/>
<replace/>
</searchReplace>
</import>
<service name="syncbpelprocess_client_ep">
<binding type="ws">
<attribute name="port">
<replace>http://xmlns.oracle.com/Application1/SyncBPELSample/SyncB
PELProcess#wsdl.endpoint(syncbpelprocess_client_ep/SyncBPELProcess
_pt)</replace>
</attribute>
</binding>
</service>
<!--Add search and replace rules for the component properties
    For components and service/reference bindings, you can add
policy references.
<component name="*">
<wsp:PolicyReferenceorawsp:category="management"
orawsp:status="enabled" URI="oracle/log_policy"/>
</component>-->
</composite>
<!--To configure monitor.config:
<property name="enabled"><replace>true</replace></property>
<property
name="dataObjectsFolder"><searchReplace><search>mydev</search><rep
lace>myproduction</replace></searchReplace></property>
    sample properties to configure for adapter:
<jca:property
name="QueueName"><replace>medmq1</replace></jca:property>

```

To add search and replace rules for wsdl's, xsd and jca files

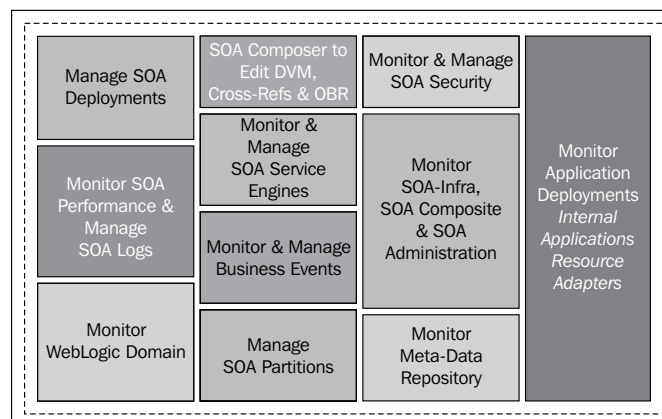
Example:

```
<searchReplace>
<search>http://my-dev-server</search>
<replace>http://my-test-server</replace>
<searchReplace>
<searchReplace>
<search>8888</search>
<replace>8889</replace>
<searchReplace>
-->
<wsdlAndSchema
name="SyncBPELProcess.wsdl|xsd/SyncBPELProcess.xsd">
<searchReplace>
<search/>
<replace/>
</searchReplace>
</wsdlAndSchema>
</SOAConfigPlan>
```

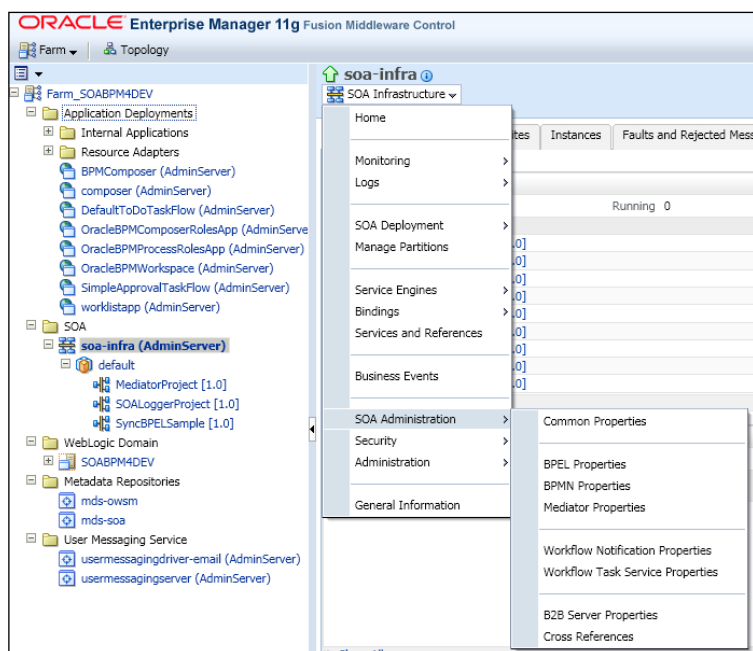
## Management of the SOA composite application using the Enterprise Manager

Oracle Enterprise Manager 11g Fusion Middleware Control provides different features to manage and monitor an SOA application and an underlying SOA infrastructure.

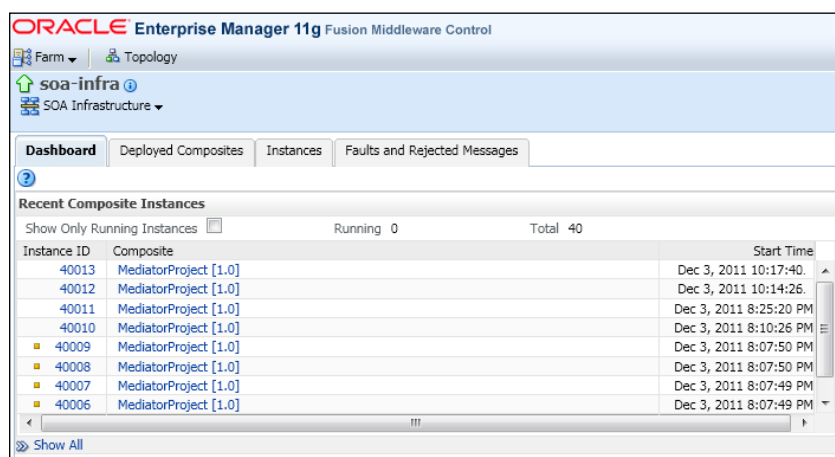
The following diagram shows the features provided by the **Fusion Middleware Control**:



The following screenshot shows the options available to manage the SOA composite application through the **SOA Infrastructure** menu:

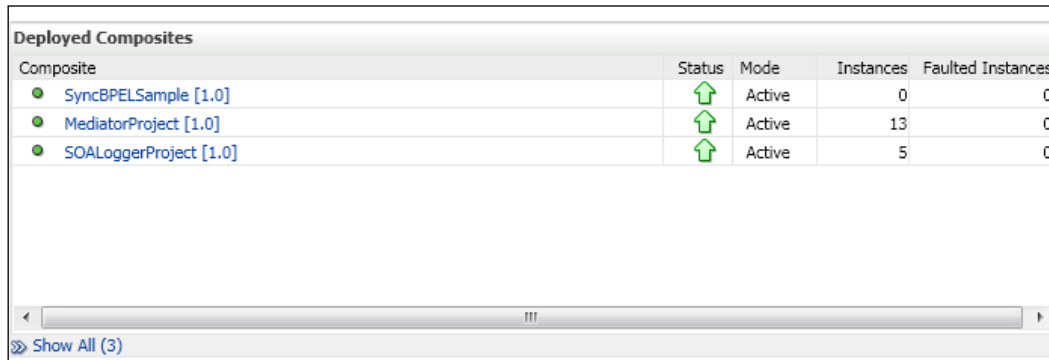


The **SOA Dashboard** provides complete access to the **SOA Infrastructure** components. **Instances** created in the SOA Suite can be monitored and viewed from the **Dashboard**. **Deployed Composites**, recent **Faults and Rejected Messages**, Service Engine metrics, and Composite Instance versus Fault Chart can also be viewed from the **SOA Dashboard**. The next two screenshots shows these details:





SCA composites can be changed to different modes from **Active** mode after deployment, for managing the transactions through the composites. The following screenshot shows the composites in **Active** mode:



The screenshot shows a window titled 'Deployed Composites' with a table listing three composites. Each composite has a green upward arrow icon in the 'Status' column, indicating it is in 'Active' mode. The 'Instances' column shows 0 for SyncBPELSample, 13 for MediatorProject, and 5 for SOALoggerProject. The 'Faulted Instances' column shows 0 for all three. At the bottom, there is a 'Show All (3)' link.

Composite	Status	Mode	Instances	Faulted Instances
● SyncBPELSample [1.0]	↑	Active	0	0
● MediatorProject [1.0]	↑	Active	13	0
● SOALoggerProject [1.0]	↑	Active	5	0

The following table shows the different modes of the composites and their significance:

Sr.No.	Mode name	Mode description	New instance can be created (Yes/No)?	Running instance will get completed (Yes/No)?	New messages will be rejected (Yes/No)
1	Active	Process incoming messages as usual.	Yes	Yes	No
2	Shutdown	Revision of the composite in this state will not initiate new instances of the composite and messages will be rejected.	No	No	Yes
3	Retire	Revision of the composite will be retired and will not initiate a new instance of the composite.	No	Yes	Yes

**Recent Faults and Rejected Messages** can be viewed from this section of the composite.







**Recent Faults and Rejected Messages**

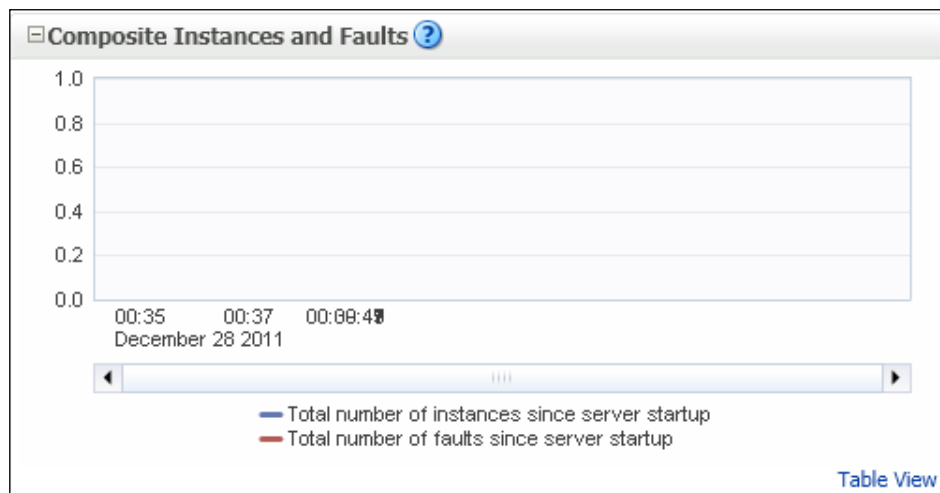
Show only system faults ☐

Error Message

No faults found

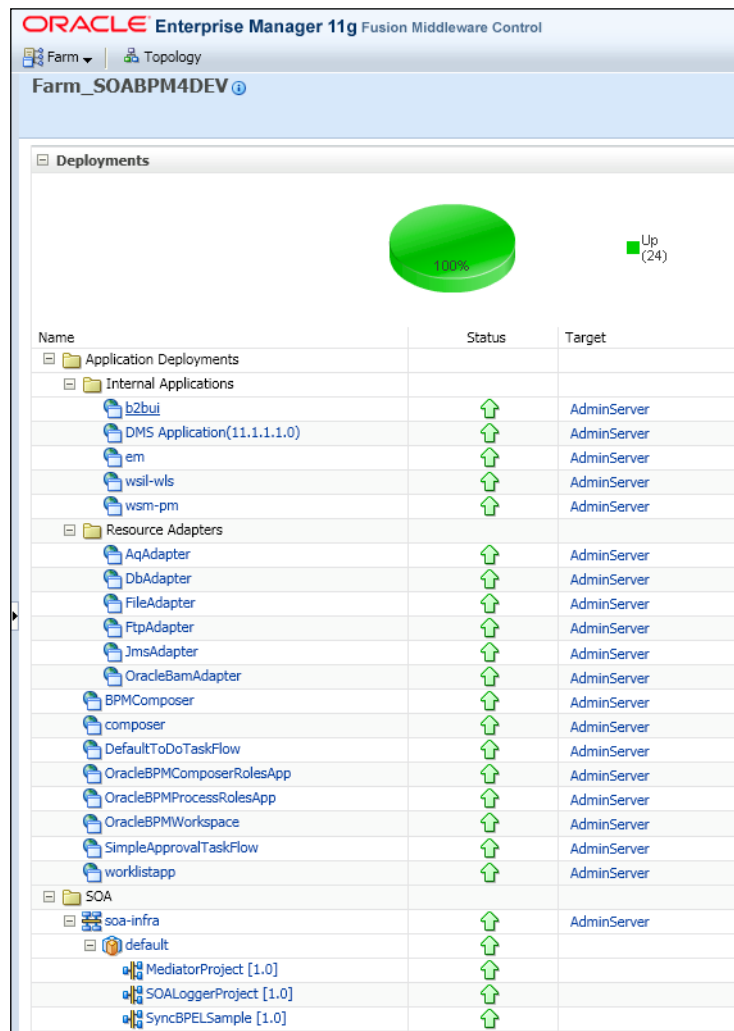
[Show All](#)

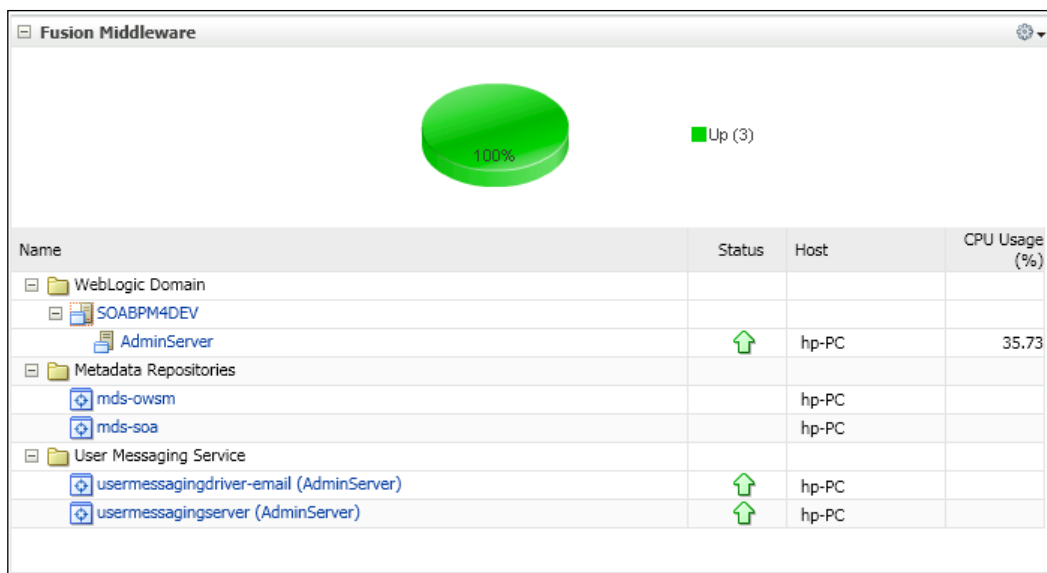
Service Engines		
Name	Number of Components	Faults
 BPEL Engine	6	0
 BPMN Engine	0	0
 Mediator Engine	4	0
 Human Workflow Engine	0	0
 Business Rules Engine	0	0
 Spring Engine	0	0



## SOA farms and topology features

SOA farms and topology views provides features to monitor the Deployed Application, Metadata Repositories, User Messaging Service, and the WebLogicDomain.





**ORACLE Enterprise Manager 11g Fusion Middleware Control**

Farm ▼ | Topology

Farm\_SOABPM4DEV

- Application Deployments
  - Internal Applications
    - Resource Adapters
      - BPMComposer (AdminServer)**
      - composer (AdminServer)
      - DefaultToDoTaskFlow (AdminServer)
      - OracleBPMComposerRolesApp (AdminServer)
      - OracleBPMProcessRolesApp (AdminServer)
      - OracleBPMWorkspace (AdminServer)
      - SimpleApprovalTaskFlow (AdminServer)
      - worklistapp (AdminServer)
  - SOA
    - soa-infra (AdminServer)
      - default
        - MediatorProject [1.0]
        - SOALoggerProject [1.0]
        - SyncBPELSample [1.0]
- WebLogic Domain
  - SOABPM4DEV
    - AdminServer
  - Metadata Repositories
    - mds-owsm
    - mds-soa
  - User Messaging Service
    - usermessagingdriver-email (AdminServer)
    - usermessagingserver (AdminServer)

**BPMComposer**

Application Deployment ▼

**Summary**

**General**

State: Active

Application Type: ear

Deployed On: AdminServer

**Servlets and JSPs**

Active Sessions: 0

Request Processing Time (ms): 0

Requests (per minute): 0.00

**Work Manager**

Requests (per minute): 0.00

Pending Requests: 0

**EJBs**

Beans in Use: 0

Bean Accesses (per minute): 0.00

Bean Access Successes (%): 0.00

Bean Transaction Commits (per minute): 0.00

Bean Transaction Rollbacks (per minute): 0.00

Bean Transaction Timeouts (per minute): 0.00

Bean Transaction Commits (%): 0.00

**Entry Points**

**Web Modules**

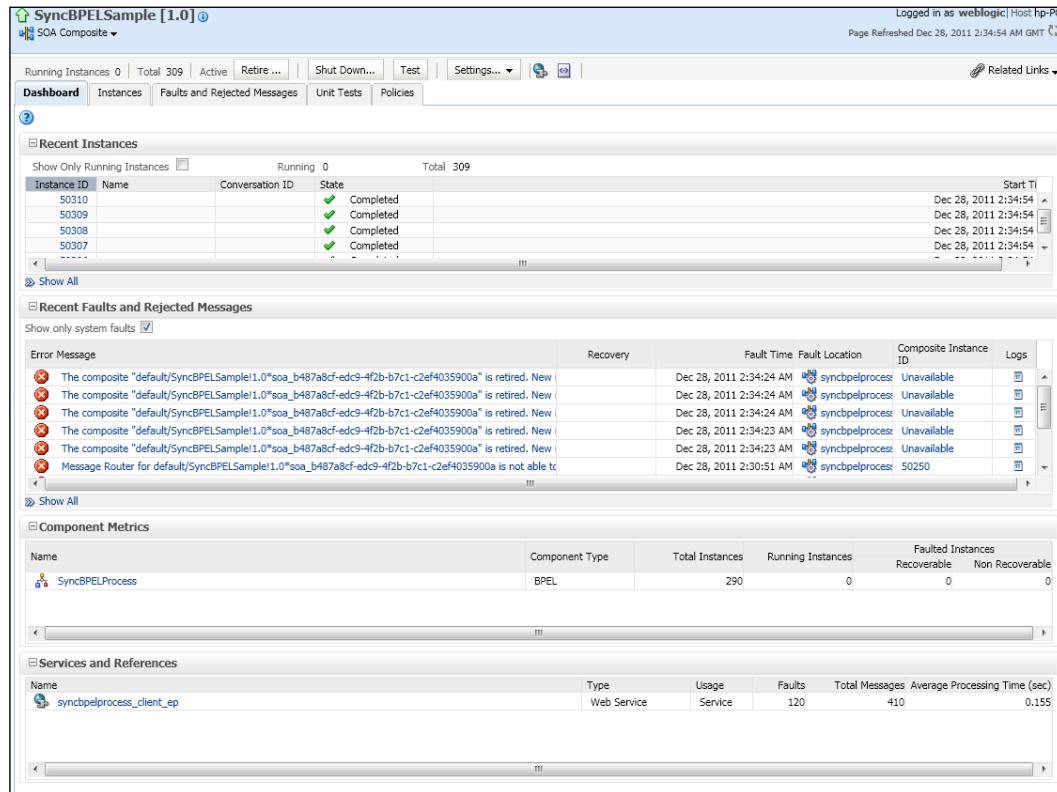
Name	Test Point
/bpm/composer	http://hp-PC:7001/bpm/composer

**Web Services**



Service Name	Port	Test
No Web Services Found		

## Managing and monitoring the SOA composite

Composite Instances, Faults, Composite Level Metrics, and Associated Services and References can be monitored and managed using the SOA EM console:



**SyncBPELSample [1.0]** Logged in as weblogic | Host: hp-PC  
Page Refreshed Dec 28, 2011 2:34:54 AM GMT

Running Instances: 0 | Total: 309 | Active: | Retire: ... | Shut Down: ... | Test: | Settings: ... |  |  | Related Links

**Dashboard** | Instances | Faults and Rejected Messages | Unit Tests | Policies

**Recent Instances**







Show Only Running Instances: ☐ Running: 0 Total: 309

Instance ID	Name	Conversation ID	State	Start Time
50310			Completed	Dec 28, 2011 2:34:54
50309			Completed	Dec 28, 2011 2:34:54
50308			Completed	Dec 28, 2011 2:34:54
50307			Completed	Dec 28, 2011 2:34:54

[Show All](#)


**Recent Faults and Rejected Messages**

Show only system faults: ☒

Error Message	Recovery	Fault Time	Fault Location	Composite Instance ID	Logs
The composite "default/SyncBPELSample!1.0"soa_b487a8cf-edc9-4f2b-b7c1-c2ef4035900a" is retired. New		Dec 28, 2011 2:34:24 AM	syncbpeprocess	Unavailable	
The composite "default/SyncBPELSample!1.0"soa_b487a8cf-edc9-4f2b-b7c1-c2ef4035900a" is retired. New		Dec 28, 2011 2:34:24 AM	syncbpeprocess	Unavailable	
The composite "default/SyncBPELSample!1.0"soa_b487a8cf-edc9-4f2b-b7c1-c2ef4035900a" is retired. New		Dec 28, 2011 2:34:24 AM	syncbpeprocess	Unavailable	
The composite "default/SyncBPELSample!1.0"soa_b487a8cf-edc9-4f2b-b7c1-c2ef4035900a" is retired. New		Dec 28, 2011 2:34:23 AM	syncbpeprocess	Unavailable	
The composite "default/SyncBPELSample!1.0"soa_b487a8cf-edc9-4f2b-b7c1-c2ef4035900a" is retired. New		Dec 28, 2011 2:34:23 AM	syncbpeprocess	Unavailable	
Message Router for default/SyncBPELSample!1.0"soa_b487a8cf-edc9-4f2b-b7c1-c2ef4035900a is not able to		Dec 28, 2011 2:30:51 AM	syncbpeprocess	50250	


[Show All](#)

**Component Metrics**

Name	Component Type	Total Instances	Running Instances	Faulted Instances Recoverable Non Recoverable
 SyncBPELProcess	BPEL	290	0	0 0


[Show All](#)

**Services and References**

Name	Type	Usage	Faults	Total Messages	Average Processing Time (sec)
 syncbpeprocess_client_ep	Web Service	Service	120	410	0.155

[Show All](#)

- A detailed view of an **Instance** with the **Search** feature is shown in the following screenshot:

 SyncBPELSample [1.0] Logged in as weblogic | Host hp-PC  
SOA Composite ▼ Page Refreshed Dec 28, 2011 2:34:54 AM GMT

Running Instances 0 | Total 309 | Active | Retire ... | Shut Down... | Test | Settings... ▼

Dashboard | **Instances** | Faults and Rejected Messages | Unit Tests | Policies











Instances of this SOA composite are listed below. There may be more instances in the database than shown in this page. Also when composite audit tracking is disabled, component instances may be created within the composite without its own instances. Click Delete with Options to purge the instances from the database.

**Search**

Instance ID  Conversation ID   
 Name  Start Time From  (UTC+00:00) London  
 ECID  Start Time To  (UTC+00:00) London

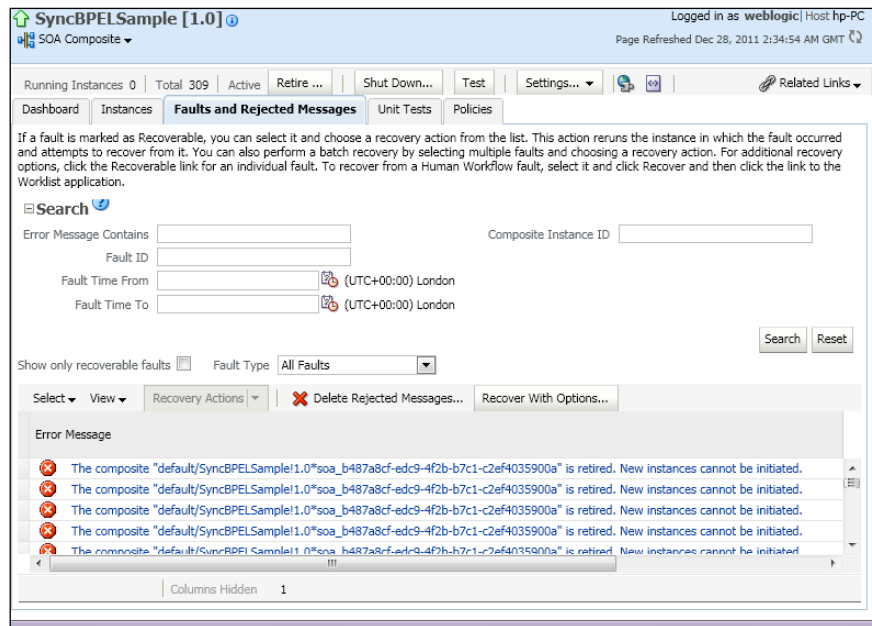
Show  ▼

View ▼

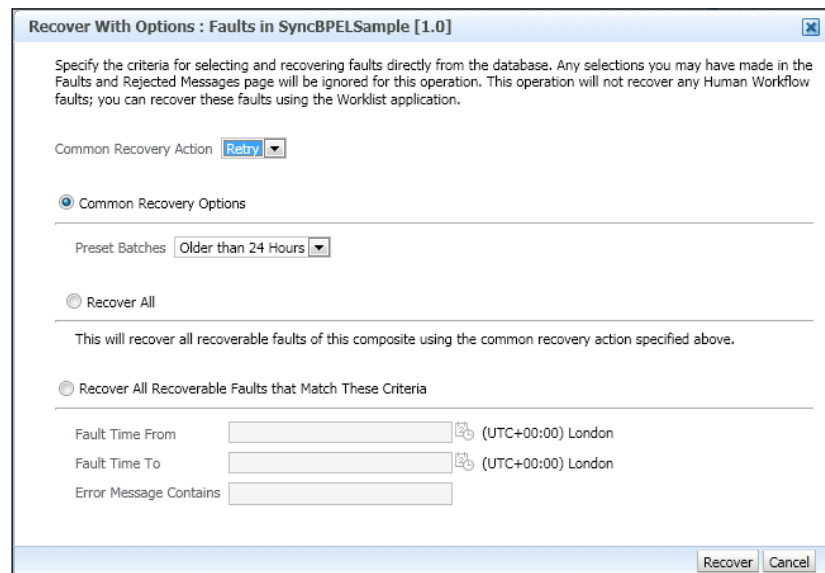
Instance ID	Name	Conversation ID	State	Start Time	Logs
50510			✓ Completed	Dec 28, 2011 2:39:56 AM	
50509			✓ Completed	Dec 28, 2011 2:39:56 AM	
50508			✓ Completed	Dec 28, 2011 2:39:56 AM	
50507			✓ Completed	Dec 28, 2011 2:39:56 AM	
50506			✓ Completed	Dec 28, 2011 2:39:56 AM	
50505			✓ Completed	Dec 28, 2011 2:39:56 AM	
50504			✓ Completed	Dec 28, 2011 2:39:56 AM	
50503			✓ Completed	Dec 28, 2011 2:39:55 AM	
50502			✓ Completed	Dec 28, 2011 2:39:55 AM	
50501			✓ Completed	Dec 28, 2011 2:39:55 AM	

Columns Hidden 2

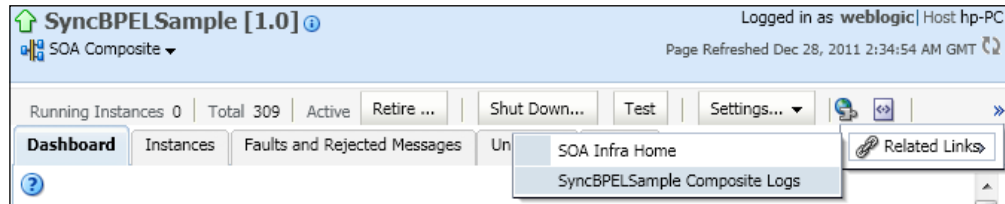
- A detailed view of **Faults and Rejected Messages**, which can be tracked using the SOA EM console is shown in the following screenshot:



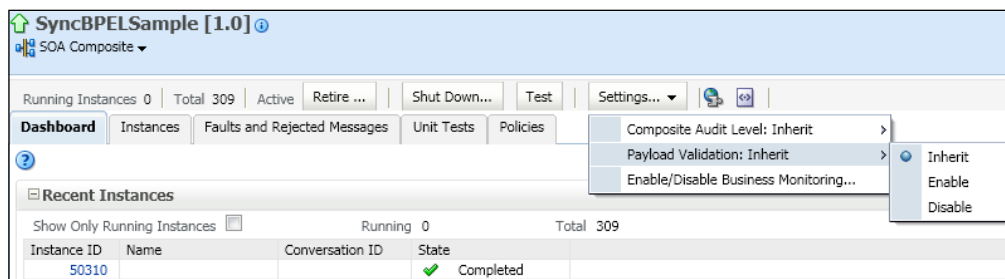
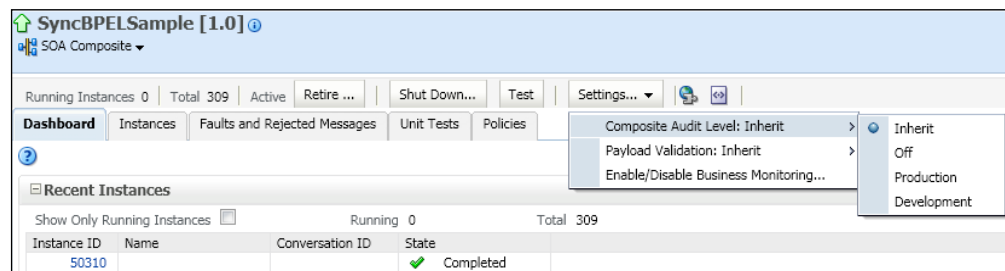
- The SOA EM console has a feature to recover the faulted instance for recoverable faults:



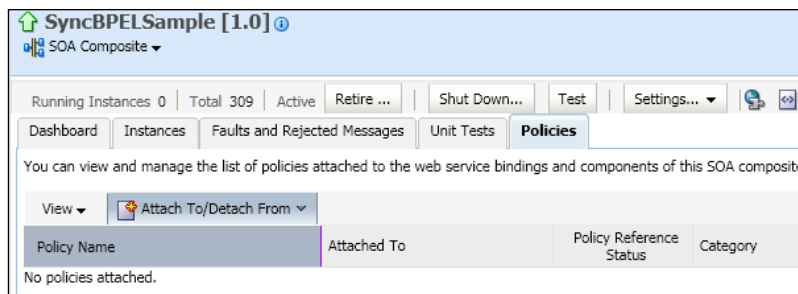
- Manage and view the Composite Level Log:



- Manage the **Composite Level Audit, Payload Validation, and Business Monitoring** settings:

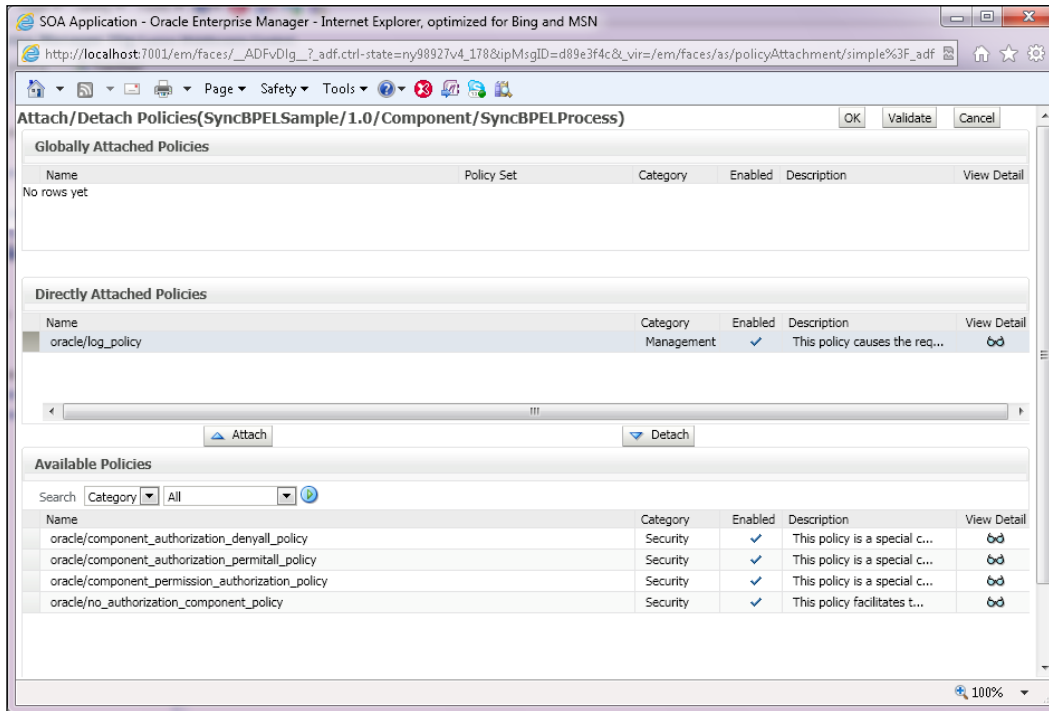


- SOA EM provides features to **Retire, Shutdown, and Test** a composite. SOA EM also provides features to attach WS-Policies dynamically during runtime:

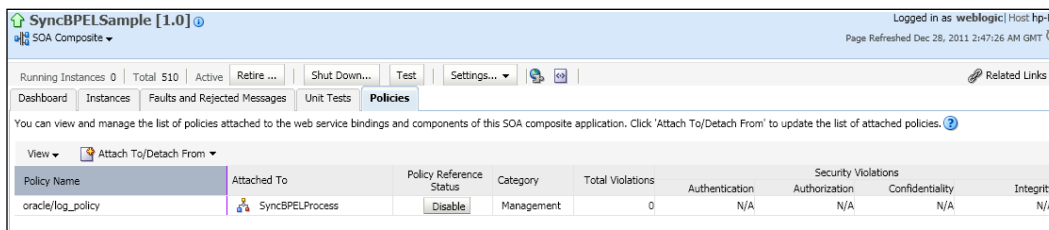




- Choose the policy and attach to the component or services:

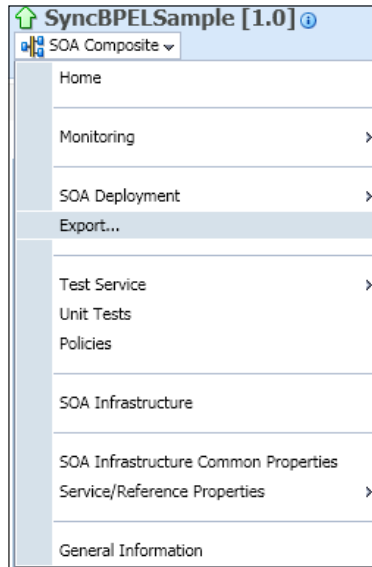


- The SOA EM console provides features to detach/attach logging, security, and reliability-related policies:



## SOA EM Export features

The Oracle SOA EM console provides features to **Export** a composite and save it as an archive. It is a very essential feature, as the composite is allowed to change at runtime, and the design-time code stored in the code base may not have all the required changes. This feature helps to manage and maintain the integrity of the SOA composite lifecycle management procedures.



## Export options

The following are the options for exporting a composite:

- **Option 1: Export with all post-deploy changes.** This option will generate a composite archive file containing the original, design-time definitions of the composite, as well as all post-deployment information listed in Options 2 and 3.
- **Option 2: Export with runtime/metadata changes only.** The composite archive file will include the original composite plus such post-deployment changes as task definitions, rule changes, and so on.
- **Option 3: Export with property changes only.** The composite archive file will include the original composite plus any post-deployment property changes such as binding properties or policy settings.

- **Option 4: Export with no post-deploy changes.** This option will generate a composite archive file containing only the pre-deployment, design-time definitions of the composite. Any property settings you may have made on a running composite, or any runtime metadata, will be ignored in the export operation.

SyncBPESample [1.0] Logged in as weblogic | Host: hp-PC  
Page Refreshed Dec 28, 2011 2:48:21 AM GMT

**Export Composite** Export Cancel

This page provides different options for exporting a snapshot of a running composite. This is useful, for example, when you want to replicate the same deployment on a different deployment target. This operation will have no effect on your currently running composite.

**You have chosen to export the following composite revision.**

Composite Name: SyncBPESample  
Composite Revision: 1.0  
Current Deployment Target: /Farm\_SOABPM4DEV/SOABPM4DEV/AdminServer/default

☒ **Option 1: Export with all post-deploy changes**  
This option will generate a composite archive file containing the original, design-time definitions of the composite; as well as all post-deployment information listed in Option 2 and 3.

☐ **Option 2: Export with runtime/metadata changes only**  
The composite archive file will include the original composite plus such post-deployment changes as task definitions, rule changes, etc..

☐ **Option 3: Export with property changes only**  
The composite archive file will include the original composite plus any post-deployment property changes, such as binding properties or policy settings.

☐ **Option 4: Export with no post-deploy changes**  
This option will generate a composite archive file containing only the pre-deployment, design-time definitions of the composite. Any property settings you may have made on a running composite, or any runtime metadata, will be ignored in the export operation.

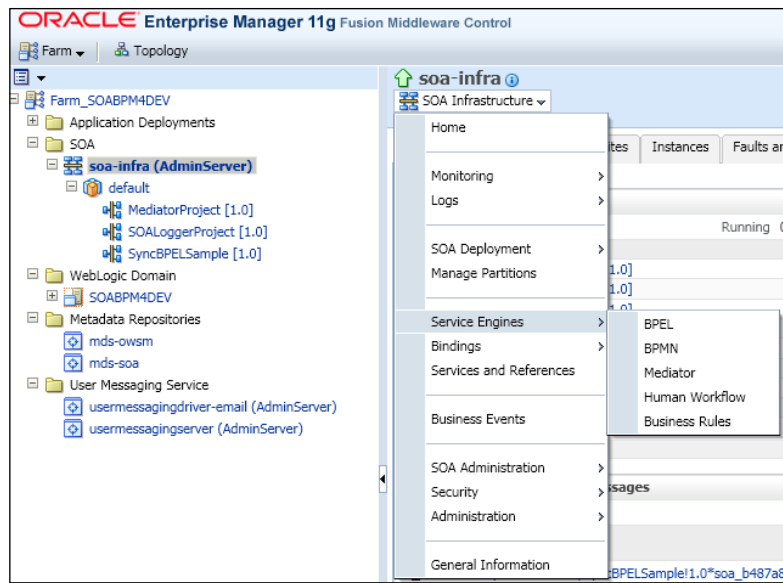
**SAR File**  
A composite archive (SAR) file will be generated with a standard name, shown below. Alternatively, you can specify your own name for the file. The file is first exported to the server where Enterprise Manager is running. When the export operation completes, you will be asked to specify a directory to save the file locally on the machine where this web browser is running. When downloading completes, click Done.

☒ Export With Default Archive Name: sca\_SyncBPESample\_rev1.0.jar

☐ Specify Custom Extension Text-EXAMPLE: sca\_SyncBPESample\_rev1.0-MyText.jar

## Monitoring SOA Service Engines

Oracle SOA Suite has five **Service Engines** as part of the **soa-infra** and it allows monitoring of the statistics such as composite instances details, faults, and monitors the statistics to understand the performance of the engine. The following screenshot shows the five **Service Engines** as part of the **soa-infra**:



## Engine-level Dashboard view

The following is a screenshot of an engine-level **Dashboard** view:

soa-infra SOA Infrastructure

Logged in as weblogic Host hp-PC  
Page Refreshed Dec 28, 2011 2:52:47 AM GMT

SOA Infrastructure Home > BPEL Engine Home

**BPEL Engine** (Service Engine) [Related Links](#)

**Dashboard** | Statistics | Instances | Faults | Deployed Components | Recovery

**Recent Instances**  
Show Only Running Instances ☐ Running 0 Total 528

Instance ID	Component	Composite	State	Start Date	Last Modified Date	Logs
bpel:50490	SyncBPELProcess	SyncBPELSample [1.0]	Completed	Dec 28, 2011 2:3...	Dec 28, 2011 2:39:56 AM	<a href="#">Log</a>
bpel:50489	SyncBPELProcess	SyncBPELSample [1.0]	Completed	Dec 28, 2011 2:3...	Dec 28, 2011 2:39:56 AM	<a href="#">Log</a>
bpel:50488	SyncBPELProcess	SyncBPELSample [1.0]	Completed	Dec 28, 2011 2:3...	Dec 28, 2011 2:39:56 AM	<a href="#">Log</a>
bpel:50487	SyncBPELProcess	SyncBPELSample [1.0]	Completed	Dec 28, 2011 2:3...	Dec 28, 2011 2:39:56 AM	<a href="#">Log</a>
bpel:50485	SyncBPELProcess	SyncBPELSample [1.0]	Completed	Dec 28, 2011 2:3...	Dec 28, 2011 2:39:56 AM	<a href="#">Log</a>

[Show All](#)

**Components**

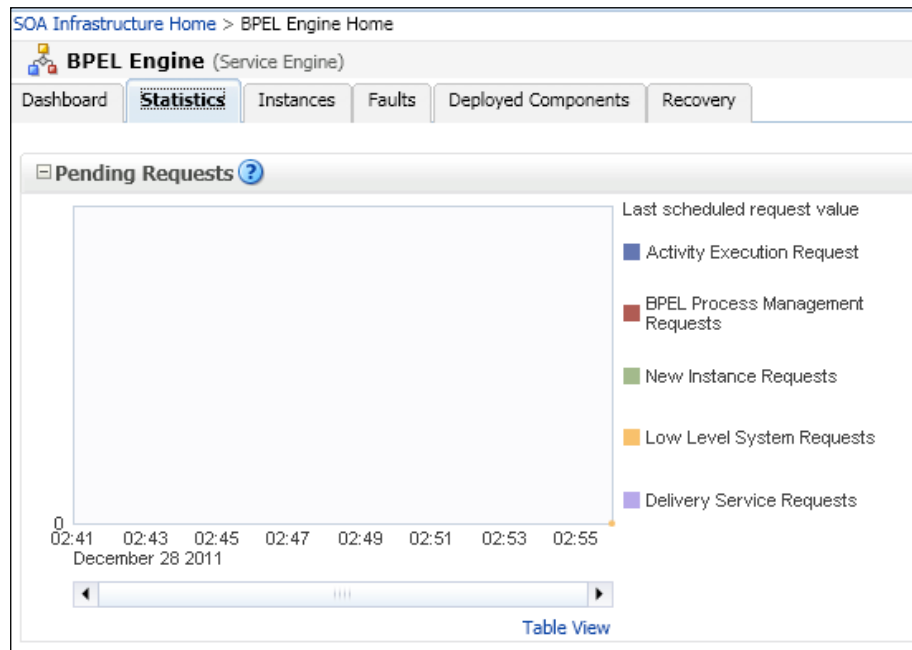
Name	Composite	Status	Total Instances	Running Instances	Faulted Instances Recoverable Non Recoverable
SyncBPELProcess	SyncBPELSample [1.0]		490	0	0 0
LogInfoThroughBPEL	SOALoggerProject [1.0]		5	0	0 0
ASync11BPELProcess	MediatorProject [1.0]		1	0	0 0
Sync11BPELProcess	MediatorProject [1.0]		7	0	0 0
Event11BPELProcess	MediatorProject [1.0]		0	0	0 0

[Show All](#)

**Recent Faults**

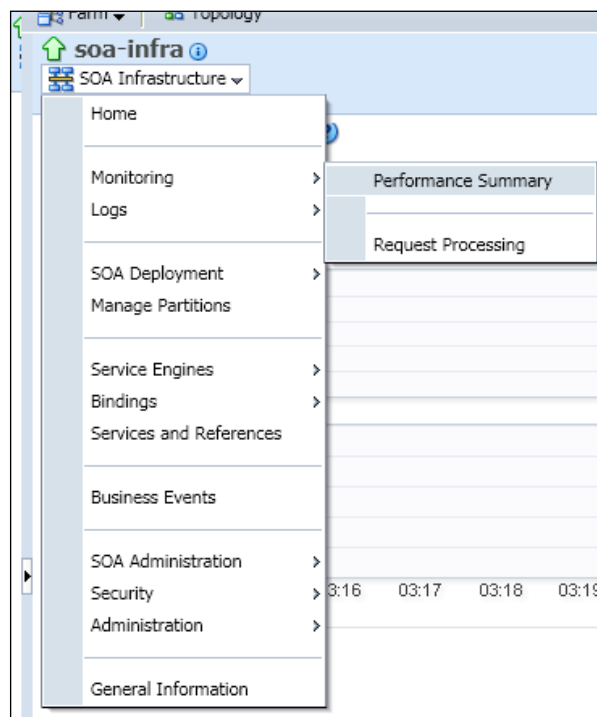
## BPEL Engine Statistics

The next two screenshots show the **BPEL Engine Statistics**:

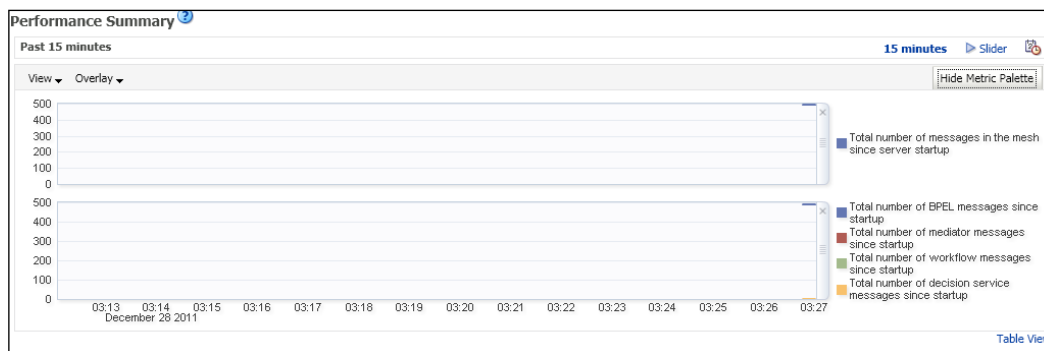


## Monitoring SOA performance

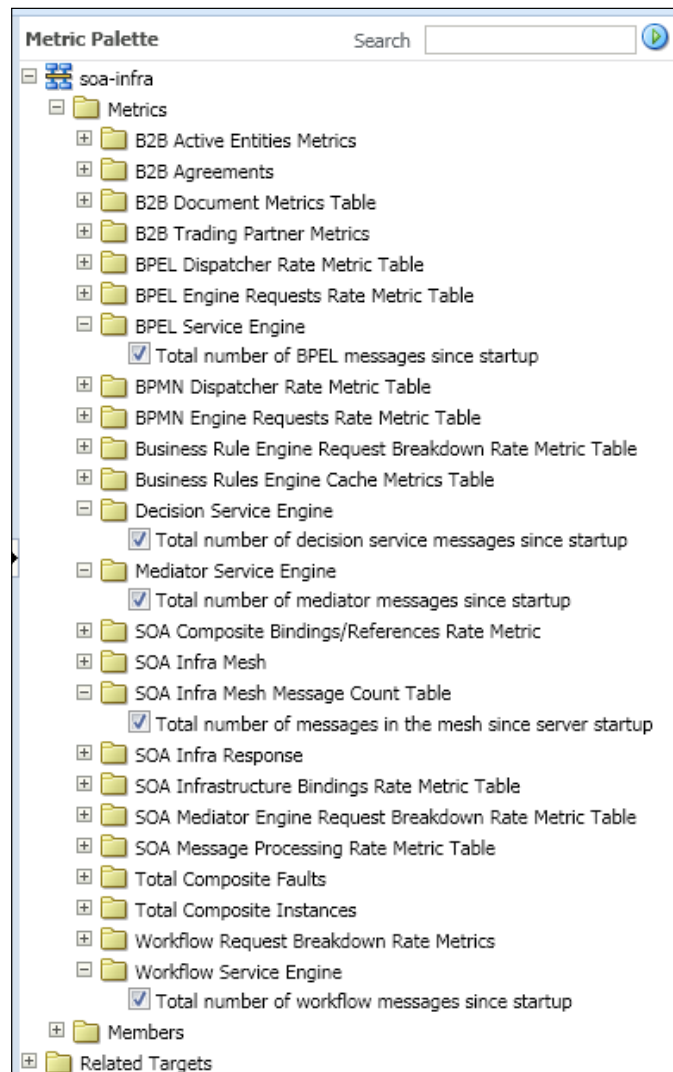
SOA performance can be monitored using the **Monitoring** option in the **SOA Infrastructure** section, as shown in the following screenshot:



The next screenshot shows the **Performance Summary** chart:

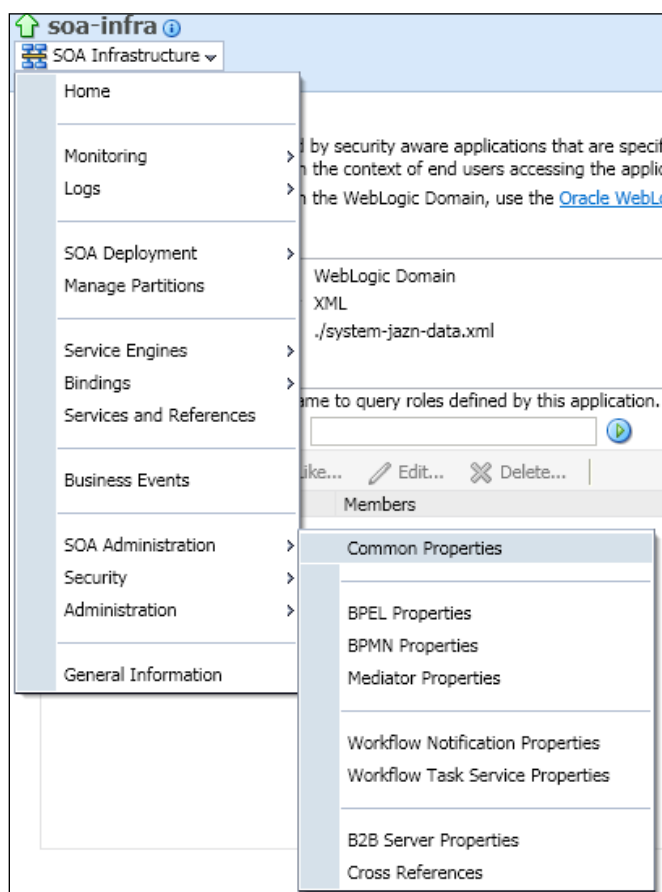


Monitoring metrics can be added using the **Metric Palette** of the **Performance Summary**, as shown in the following screenshot:



## Managing soa-infra and Service Engine properties

The SOA EM console provides the features to manage common properties and service engine-level properties, as shown in the following screenshot:





**SOA Infrastructure Common Properties** such as **Audit Level** and **Payload Validation** can be managed using this section, as shown in the following screenshot:

The screenshot shows the 'SOA Infrastructure Common Properties' configuration page. The page is titled 'SOA Infrastructure Home > Common Properties' and 'SOA Infrastructure Common Properties'. It includes a 'Related Links' section. The main configuration area is divided into several sections: 'Audit Level' (set to 'Development'), 'Capture Composite Instance State' (checked), 'Payload Validation' (unchecked), 'UDDI Registry Properties' (Inquiry URL, User, Password), 'Server URLs' (Callback Server URL, Server URL), and 'Data Display Options' (Duration Details: 24 hours). A right-hand panel titled 'Configuring the Common SOA Infrastructure Properties' provides detailed descriptions for 'Audit Level' (Production, Development, Off), 'Capture Composite Instance State', and 'Payload Validation'. An 'Advanced' section is also visible at the bottom.

**BPEL Engine Properties** can be managed using the following section:

The screenshot shows the 'BPEL Service Engine Properties' configuration page. The page is titled 'SOA Infrastructure Home > BPEL Properties' and 'BPEL Service Engine Properties'. It includes a 'Properties' section with the instruction 'Edit property values and click Apply.' The main configuration area lists several properties: '\* Audit Level' (set to 'Development'), '\* Audit Trail Threshold (Byte)' (50000), '\* Large Document Threshold (Byte)' (100000), '\* Dispatcher System Threads' (2), '\* Dispatcher Invoke Threads' (20), '\* Dispatcher Engine Threads' (30), '\* Payload Validation' (unchecked), and '\* Disable BPEL Monitors and Sensors' (unchecked). A 'More BPEL Configuration Properties...' link is at the bottom.

**BPMN Engine Properties** can be managed using the following section:

**BPMN Service Engine Properties**

**Properties**

Edit property values and click Apply.

\* Audit Level
Development

\* Audit Trail Threshold (Byte)
50000

\* Large Document Threshold (Byte)
100000

\* Dispatcher System Threads
2

\* Dispatcher Invoke Threads
20

\* Dispatcher Engine Threads
30

\* Payload Validation
☐

\* Disable BPMN Monitors and Sensors
☐

[More BPMN Configuration Properties...](#)

**Mediator Engine Properties** can be managed using the following section:

[SOA Infrastructure Home](#) > Mediator Properties
**Mediator Service Engine Properties**

**Properties**

Edit property values and click Apply to save the changes.

Audit Level
Inherit

Metrics Level
Enabled

Parallel Worker Threads
4

Parallel Maximum Rows Retrieved
200

Parallel Locker Thread Sleep(sec)
2

Error Locker Thread Sleep(sec)
5

Parameters

Container ID Refresh Time(sec)
60


Container ID Lease Timeout(sec)
300

Resequencer Locker Thread Sleep(sec)
10

Resequencer Maximum Groups Locked
4

Resequencer Worker Threads
4

[More Mediator Configuration Properties...](#)

 **soa-infra** ⓘ  
SOA Infrastructure ▼

SOA Infrastructure Home > Workflow Notification Properties

**Information** ⓘ  
All changes made in this page require a server restart to take effect.

**Workflow Notification Properties** ⓘ  
Before configuring the Workflow Notification, configure the Messaging Service Driver. [Go to the Messaging Driver page](#)

\* Notification Mode

**Notification Service**

\* Email : From Address \*

\* Email : Actionable Address \*

\* Email : Reply To Address \*

[More Workflow Notification Configuration Properties...](#)

SOA Infrastructure Home > Workflow Task Service Properties

**Workflow Task Service Properties** ⓘ

Related Links ⓘ

\* Actionable Email Account

\* Workflow Service Session Timeout (in minutes)

Workflow Custom Classpath URL

**Dynamic Assignment and Task Escalation Functions**

View ▼

Function Name	Classpath
MANAGERS_MANAGER	oracle.bpel.services.workflow.assignment.dynamic.patterns.TaskEscalationManagersManager
ROUND_ROBIN	oracle.bpel.services.workflow.assignment.dynamic.patterns.RoundRobin
LEAST_BUSY	oracle.bpel.services.workflow.assignment.dynamic.patterns.LeastBusy
<b>MOST_PRODUCTIVE</b>	oracle.bpel.services.workflow.assignment.dynamic.patterns.MostProductive

**Parameters: MOST\_PRODUCTIVE**

Name	Value
DEFAULT_TIME_PERIOD	<input type="text" value="7"/>

ⓘ

\* Worklist Application URL

\* Pushback Assignee

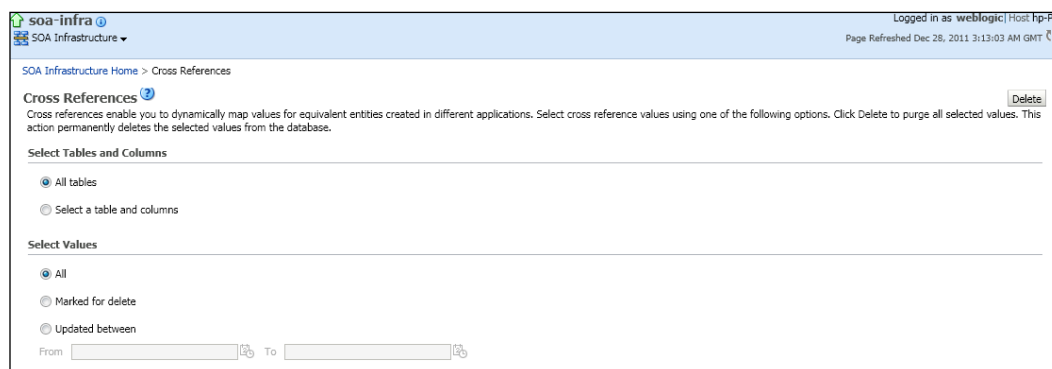
\* Portal Realm Mapping

**Task Auto Release Configuration**

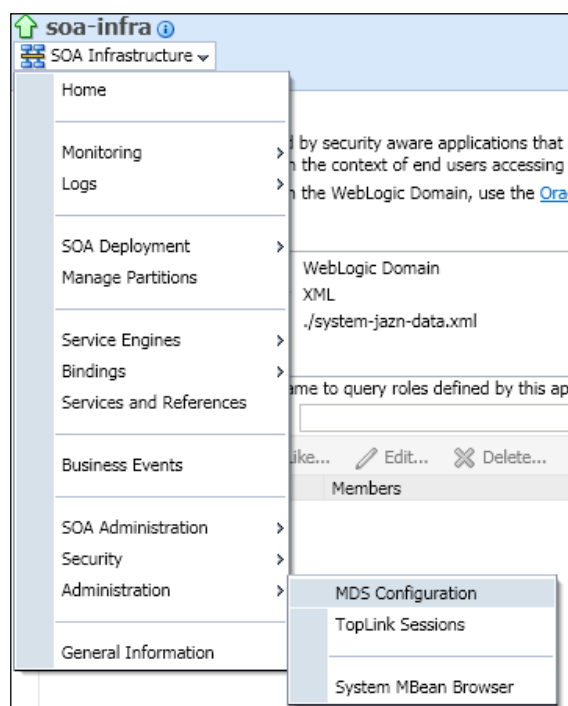
Priority	Default Duration	Percentage of Expiration
5	<input type="text" value="PSD"/>	<input type="text" value="70"/>
4	<input type="text" value="P4D"/>	<input type="text" value="60"/>
3	<input type="text" value="P3D"/>	<input type="text" value="50"/>
1	<input type="text" value="P1D"/>	<input type="text" value="30"/>

[More Workflow Taskservice Configuration Properties...](#)

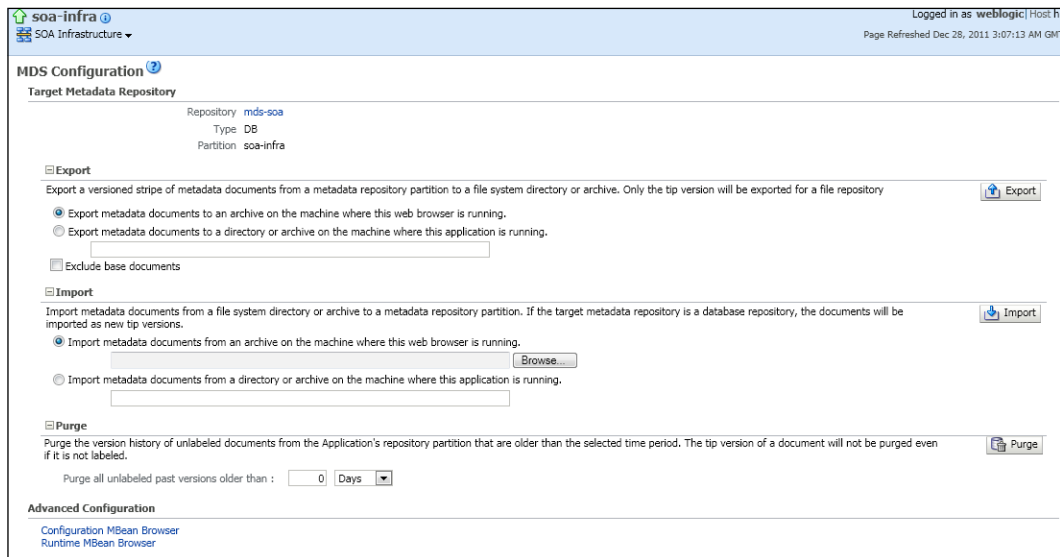
Oracle SOA EM provides features to manage **Cross References** as well, as shown in the following screenshot:



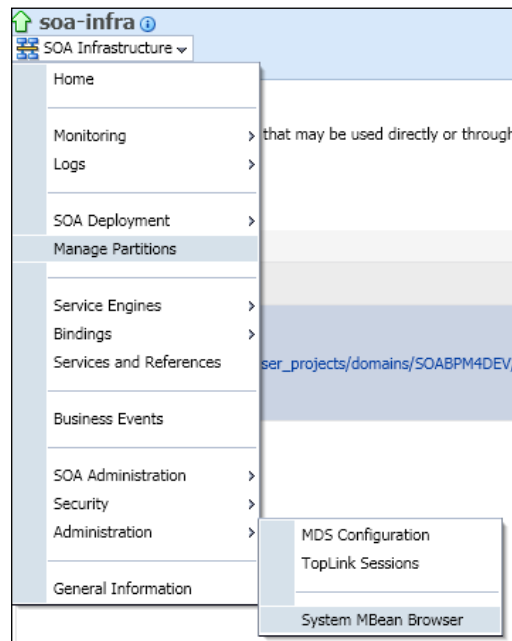
Oracle SOA EM provides features to manage **MDS Configuration** and modify system-level properties through the **MBean Browser**, as shown in the following screenshot:



**MDS Repository** can be exported, imported, or purged using the features shown in the following screenshot:



**MBean Browser** is a powerful tool to manage the properties exposed through JMX API of the SOA composite components:



The advanced properties of **soa-infra** are shown in the following screenshot:

Configuration MBeans: AppDeployment:soa-infra

Name	Description	Access	Value
1 AbsoluteInstallDir	The fully resolved location of this application's installation root directory on the Administration Server.	R	
2 AbsolutePlanDir	The fully resolved location of this application's deployment plan directory on the Administration Server.	R	
3 AbsolutePlanPath	The fully resolved location of this application's deployment plan on the Administration Server.	R	
4 AbsoluteSourcePath	The fully resolved location of this application's source files on the Administration Server.	R	C:\Oracle\Middleware\Ora
5 ApplicationIdentifier	The Application Identifier of the application version uniquely identifies the application version across all versions of all applications. If the application is not versioned, the Application Identifier is the same as the application name.	R	soa-infra
6 ApplicationName	The name of the application. Note that the name of the current MBean is not the name of the application.	R	soa-infra
7 CompatibilityName	This is only set for beans created as a result of conversion from an 8.1 application configured using ApplicationMBean and ComponentMBean.	RW	
8 DeploymentOrder	An integer value that indicates when this unit is deployed, relative to other deployable units on a server, during startup. Units with lower values are deployed before those with higher values.	RW	350
9 DeploymentPlan	The contents of this application's deployment plan, returned as a byte[] containing the XML.	R	
10 DeploymentPlanExternalDescriptor	A zip file containing the external descriptors referenced in the deployment plan.	R	
11 DeploymentPrincipalName	A string value that indicates what principal should be used when deploying the file or archive during startup and shutdown	RW	
12 InstallDir	The path to application's install-root directory, relative to the domain/config/deployments directory.	R	
13 ModuleType	The values match those defined by Jsr88. This attribute may	RW	ear

## The Enterprise Manager Grid Control in the SOA Management Packs

The Enterprise Manager Grid Control in the SOA Management Packs provides the following features:

- Provides features to automate the SOA Management and creates alerts and reports for the administration
- Provides features to enable right software configuration management techniques, and provides continuous integration through the service life cycle management
- Provides features to automate the deployments of SOA components through windows and schedule-based deployment to different environments
- Provides features to generate views for composite history analysis and reports for various metrics such as service usage details, service-level availability indicator, and so on

- Provides enterprise visibility into business transactions across different Oracle SOA tiers
- Monitors the performance of Oracle SOA/SCA components, Spring, and OSB
- Provides features to compare BPEL, the Oracle Service Bus server, and domain configuration parameters with other servers and domains
- Provides features to compare the BPEL process versions to observe changes across different versions of deployments
- Provides features to clone SCA composites directly from the test to the production environment
- Provides features to clone from fully tested golden images of the SCA composite from software configuration management/software version control solutions
- Makes provisions for new composites or new versions of existing composites to existing WebLogic domains
- Provides features to specify configuration plans for customizing the reference during deployment to different sets of environments
- Provides agility to the enterprise through out-of-the-box SOA composite management

## Summary

In this chapter, we have discussed in detail about the following list of exam objectives:

- **Describe deploy and undeploy the SOA composite application:** We identified the various options available to deploy and undeploy the SOA composites through console, JDeveloper, WSLT, and Ant scripts
- **Describe management of the SOA composite application using the Enterprise Manager:** We understood the features supported and provided by Oracle Enterprise Manager to manage the life cycle of the SCA composite

## Self-review questions

1. SOA composites can be deployed from the console.
  - a. True
  - b. False
2. SOA composites can be exported to a file.
  - a. True
  - b. False
3. User ID and password is required to deploy a composite into SOA Runtime.
  - a. True
  - b. False
4. SCA composites can be deployed using a \_\_\_\_\_ file.
  - a. build.xml
  - b. deployconfig.xml
  - c. deploy.xml
  - d. ant-sca-deploy.xml
5. Customizations to a SAR file cannot be done from the deployment executed through the console.
  - a. True
  - b. False
6. SAR expands to \_\_\_\_\_.
  - a. Service Achieve Repository
  - b. Service Archive Repository
  - c. Service Archive
  - d. Standard Archive Repository



7. Cross references can be created from the console.
  - a. True
  - b. False
8. Domain-value maps cannot be created and exported to a file.
  - a. True
  - b. False
9. Business Rules cannot be edited from the console.
  - a. True
  - b. False
10. The deployment of composites can be controlled by a specific user.
  - a. True
  - b. False
11. B2B is a service engine in Oracle SOA Suite 11g.
  - a. True
  - b. False
12. BPEL code cannot be viewed from the console.
  - a. True
  - b. False
13. By shutting down the composite, the running instance will be terminated.
  - a. True
  - b. False
14. By retiring a composite, new instances cannot be initiated.
  - a. True
  - b. False
15. Oracle SOA Suite allows us to only test the primary service exposed in a composite.
  - a. True
  - b. False

16. Overriding the composite definition through re-deployment changes the status of the completed instance to stale.
  - a. True
  - b. False
17. Composites cannot be undeployed from the SOA EM console.
  - a. True
  - b. False
18. Additional authentication is required to change composite properties.
  - a. True
  - b. False
19. The console provides a features to add runtime properties for a composite through:
  - a. Editing `composite.xml` from console
  - b. Only **MBean Browser** can be used to change the properties
  - c. We cannot define runtime properties for a composite
  - d. SCA specifications don't support this
20. Security Policies can be added to a composite at runtime in Oracle SOA Suite 11g.
  - a. True
  - b. False
21. Rejected and faulted messages can be viewed from the console.
  - a. True
  - b. False
22. Publication and subscription of Business Events can be monitored using the console.
  - a. True
  - b. False

## **Additional references**

To learn more about monitoring SOA composites using Enterprise Manager Grid Control exam objectives, please go through the following YouTube videos which give a detailed view of the features of Oracle Enterprise Grid Control to monitor the SOA composites.

**Video title:** Monitoring SOA infrastructure components using Enterprise Manager Grid Control

**Video link:** [http://www.youtube.com/watch?v=kdtfUZc-T\\_o](http://www.youtube.com/watch?v=kdtfUZc-T_o)

**Video title:** Monitoring SOA composites using Enterprise Manager Grid Control

**Video link:** <http://www.youtube.com/watch?v=an0Rw0ZLAM0>



# Pre-assessment Test

This section of the book will provide you with an opportunity to test your knowledge in Oracle SOA Suite before going through this book.

1. BPEL expands to:
  - a. Business Process Execution Language
  - b. Business Process Expression Language
  - c. Business Process Enablement Language
  - d. Business Programming Expression Language
2. WSDL should always contain the SOAP address.
  - a. True
  - b. False
3. XQuery can be used in BPEL.
  - a. True
  - b. False
4. XSL is not supported in BPEL 2.0.
  - a. True
  - b. False
5. UDDI implementation can be achieved through:
  - a. OSR – Oracle Service Registry
  - b. OER – Oracle Enterprise Registry
  - c. MDS – Metadata Repository
  - d. Oracle Enterprise Manager

6. Executable BPEL is a/an \_\_\_\_\_ file.
  - a. XML
  - b. Text
  - c. Java
  - d. Binary
7. Abstract BPEL is a/an \_\_\_\_\_ file.
  - a. XML
  - b. Text
  - c. Java
  - d. Binary
8. All Oracle adapters are JCA compliant.
  - a. True
  - b. False
9. Composite definition is stored in the \_\_\_\_\_ file.
  - a. components.xml
  - b. composite\_details.xml
  - c. composite.xml
  - d. components.details
10. Execution of Java Callout is supported in \_\_\_\_\_. Choose the correct answer from the following list:
  - a. Mediator
  - b. BPEL
  - c. Business Rules
  - d. Human Workflow
11. Mediator should be rebuilt if the underlying WSDL definition for the Mediator is changed.
  - a. True
  - b. False
12. JDeveloper provides an option to attach WS-Policies.
  - a. True
  - b. False

13. A Composite application consists of:
- a. Services
  - b. References
  - c. Wires
  - d. All the above
14. Identify which of the following BPEL constructs terminates the process:
- a. `<terminate/>`
  - b. `<exit/>`
  - c. `<end/>`
  - d. `<teminateAndExit/>`
15. Identify which of the following BPEL constructs dehydrates the BPEL process:
- a. `<wait/>`
  - b. `<receive/>`
  - c. `<pick/>`
  - d. `<bpelx:dehydrate/>`
  - e. All the above
16. While is an Oracle extension to the BPEL construct.
- a. True
  - b. False
17. Transformation is an Oracle extension to the BPEL construct.
- a. True
  - b. False
18. Identify the Oracle extensions added to the Oracle BPEL constructs; choose the right answers from the following list:
- a. Replay
  - b. Validate
  - c. E-mail
  - d. Assign

19. JDeveloper provides a unit test framework for all components listed below BPEL, Mediator, XSL, Business Rules, Human Workflow.
  - a. True
  - b. False
20. JDeveloper provides a unit test framework to assert and emulate the BPEL process execution.
  - a. True
  - b. False
21. Business policies are configured in:
  - a. Business Rules
  - b. Cross references
  - c. Domain-value maps
  - d. BPEL code
  - e. None of the above
22. Business Rules deployed from the JDeveloper cannot be edited.
  - a. True
  - b. False
23. Composites cannot be undeployed from the enterprise manager directly.
  - a. True
  - b. False
24. By shutting down the composite, the running instance will be terminated.
  - a. True
  - b. False
25. Which of the following activities are supported in fault management of the BPEL process? Choose the right options from the following list:
  - a. Catch
  - b. CatchAll
  - c. Throw
  - d. Compensate
  - e. Assign

26. DVM and cross references can be edited using \_\_\_\_\_.  
a. SOA Composer  
b. Oracle Composer  
c. Enterprise Manager Console  
d. SOA Console
27. Event definitions are stored in \_\_\_\_\_.  
a. Event definition files with an .edl extension  
b. Metadata repository  
c. Oracle SOA database  
d. None of the above
28. Binding faults can be caught using:  
a. CatchAll  
b. Catch  
c. Fault handler block  
d. Main scope only
29. The SCA Composite definition file is packaged as a/an \_\_\_\_\_ for deployment into the console.  
a. SAB file  
b. SAR file  
c. JAR file  
d. MAR file
30. B2B is a service engine in Oracle SOA Infrastructure.  
a. True  
b. False





# B

## Post-assessment Test

This section of the book will provide an opportunity to test your knowledge in Oracle SOA Suite. We have provided a full-length test paper in this chapter for an exam experience.

1. SCA supports implementation using all the programming languages except:
  - a. Java
  - b. PHP
  - c. C++
  - d. Groovy
2. Service Component Architecture specification doesn't include:
  - a. SCA Policy Framework
  - b. SCA JMS Binding
  - c. SCA JCA Framework
  - d. None of the above
3. All types of Mediator faults can be managed through Fault Policies.
  - a. True
  - b. False
4. Multiple input parameters can be used in XSLs for transformation of messages in SOA Suite 11g.
  - a. True
  - b. False

5. DB connection to MDS is required to create an MDS connection.
  - a. True
  - b. False
6. The behavior of BPEL Dehydration cannot be altered.
  - a. True
  - b. False
7. XSLs cannot be referred to MDS.
  - a. True
  - b. False
8. Identify the standards that enable SOA in Oracle SOA Suite 11g.
  - a. SDO
  - b. JMS
  - c. JCA
  - d. SCA
  - e. All the above
9. The key benefits of SCA include:
  - a. Efficiency
  - b. Productivity
  - c. Easy maintenance
  - d. Loose coupling
  - e. All the above
10. The SOA fault policy can be configured from the SOA Composite Editor.
  - a. True
  - b. False
11. Internal messaging transport is provided by \_\_\_\_\_ in Oracle SOA Suite 11g.
  - a. Binding components
  - b. Service infrastructure
  - c. Wires
  - d. Service engines

- 
12. Analyzing existing applications and assets to identify those that can be used as services is a top-down approach in implementing SOA.
- a. True
  - b. False
13. Identify the binding components from the following list:
- a. HTTP Binding
  - b. JCA Adapters
  - c. BAM Adapters
  - d. ADF-BC Services
  - e. All the above
  - f. None of the above
14. Execution of the Java callout is supported in \_\_\_\_\_. Choose the correct options from the following components:
- a. Mediator
  - b. BPEL
  - c. Business Rules
  - d. Human Workflow
15. If the underlying WSDL definition for the mediator is changed, it is required to rebuild the Mediator.
- a. True
  - b. False
16. The following WSDL content represents a:
- ```
<wsdl:portType name="BPELProcess">
  <wsdl:operation name="process">
    <wsdl:input message="client:BPELProcessRequestMessage" />
    <wsdl:output message="client:BPELProcessResponseMessage" />
  </wsdl:operation>
</wsdl:portType>
```
- a. Synchronous BPEL
  - b. Asynchronous BPEL
  - c. One-way BPEL
  - d. None of the above

17. A composite application consists of:
- Services
  - References
  - Wires
  - All the above
18. Identify which of the following BPEL constructs terminates the process.
- `<terminate/>`
  - `<exit/>`
  - `<end/>`
  - `<teminateAndExit/>`
19. Identify which of the following BPEL constructs dehydrates the BPEL Process.
- `<wait/>`
  - `<receive/>`
  - `<pick/>`
  - `<bpelx:dehydrate/>`
  - All the above
20. Pick is an Oracle Extension to a BPEL construct.
- True
  - False
21. Validation of an XML message received by the SOA-infra layer can be enabled by:
- Using Validate Activity in a BPEL process
  - Using the Validate XSD option in the Mediator
  - Using Schematron Files in the Mediator
  - All the above
  - None of the above
22. Assert is an Oracle Extension to a BPEL construct.
- True
  - False

23. Identify the Oracle Extensions added to Oracle BPEL constructs; choose the right options from the following:
- a. Signal
  - b. IM
  - c. Java Embedding
  - d. SMS
  - e. All the above
24. A synchronous BPEL process can indefinitely wait for the response.
- a. True
  - b. False
25. Adding a `wait` activity to a synchronous process will result in compilation errors.
- a. True
  - b. False
26. Transactionality of the service component can be specified using:
- a. `bpel.configuration.transaction`
  - b. `bpel.config.transaction`
  - c. `bpel.config.transactionality`
  - d. None of the above
27. `SyncMaxWaitTime` is applicable to a synchronous BPEL process in all scenarios.
- a. True
  - b. False
28. Mid-Process receive should have the `createInstance` attribute set to `true`.
- a. True
  - b. False
29. By shutting down the composite, the running instance will be terminated.
- a. True
  - b. False

30. Which of the following activities are supported in Fault Management of the BPEL process; choose the right options.
- a. Catch
  - b. CatchAll
  - c. Throw
  - d. Compensate
  - e. Assign
31. DVM and Cross References can be edited using \_\_\_\_\_.
- a. SOA composer
  - b. Oracle composer
  - c. Enterprise Manager Console
  - d. SOA console
32. SCA Composite definition supports fail-over configuration for the web services location.
- a. True
  - b. False
33. Binding Faults can be caught using:
- a. CatchAll
  - b. Catch
  - c. Fault Handler block
  - e. Main scope only
34. The SCA Composite definition file is packaged as \_\_\_\_\_ for deployment into the console.
- a. SAB file
  - b. SAR file
  - c. JAR file
  - d. MAR file
35. XSLs can be tested from JDeveloper.
- a. True
  - b. False

36. Oracle SOA Suite supports BPEL 2.0 IMA (Inbound Message Activity).
- a. True
  - b. False
37. Dynamic Partner Links are supported in BPEL 2.0.
- a. True
  - b. False
38. Identify the activities that support synchronizing flow activities; choose the best option from the following:
- a. <sources>
  - b. <targets>
  - c. <source>
  - d. <target>
  - e. All the above
39. for-each, a BPEL 2.0 construct, supports parallel execution of each branch.
- a. True
  - b. False
40. Which of the following constructs are supported by BPEL 2.0; choose the best option.
- a. If-elseif-else
  - b. While
  - c. Repeat-Until
  - d. For-Each
  - e. All the above
41. CompensateScope can be used in which of the following activities; choose the right options.
- a. Main scope
  - b. Catch block
  - c. CatchAll block
  - d. All the above



42. The `skipCondition` attribute of the assignment statement is supported in BPEL 2.0.
- a. True
  - b. False
43. `joinFailure` is a standard BPEL Fault.
- a. True
  - b. False
44. The Fault Policy file and the Fault Binding file should always be placed in the same directory as `composite.xml`.
- a. True
  - b. False
45. `getFaultAsString()` function can be used in the Catch block to retrieve the fault information.
- a. True
  - b. False
46. `bpel.config.transaction` property does not apply to mid-process receive activities.
- a. True
  - b. False
47. BPEL and Mediator can be tested from JDeveloper.
- a. True
  - b. False
48. The following snippet of code creates a new transaction:
- ```
<component name="BPELProcess">
<implementation.bpelsrc="BPELProcess.bpel"/>
    <property name="bpel.config.transaction" many="false"
type="xs:string">required</property>
</component>
```
- a. True
  - b. False

49. When BPELCaller calls BPELCallee that has `bpel.config.transaction` set to `requiresNew`, which of the following statements is true?
- a. BPELCallee transaction is saved when BPELCallee returns a successful response or replies to a fault message that can be handled by the BPELCaller
  - b. BPELCallee transaction is rolled back when BPELCallee throws a fault that cannot be handled by the BPELCaller
  - c. BPELCallee transaction is rolled back when BPELCallee throws a rollback fault.
  - d. All the above
50. In-flight BPEL process will not persist the messages from the transaction.
- a. True
  - b. False
51. Oracle Service Bus provides which of the following features?
- a. Service Virtualization
  - b. Content-Based Routing
  - c. Service Chaining
  - d. All the above
52. Pipeline pair represent Request-Response service pattern.
- a. True
  - b. False
53. Oracle B2B enqueues error messages into `IP_IN_QUEUE`.
- a. True
  - b. False
54. MFL is used in Oracle Service Bus for:
- a. Assignments
  - b. Data manipulation
  - c. Data transformation
  - d. All the above

55. Oracle Service Bus supports which of the following transport protocol?
- a. EJB/RMI
  - b. SFTP
  - c. MQ
  - d. All the above
56. Which of the following OSB patterns help to achieve the parallel execution of the services?
- a. VETO
  - b. Fan-Out
  - c. Split-Join
  - d. None of the above.
57. BAM sensor information can be published using:
- a. Database
  - b. JMS Queue
  - c. JMS Topic
  - d. All the above
58. Oracle Business Rules can be coded as:
- a. If-then rules
  - b. Decision tables
  - c. Java expression facts
  - d. All the above
59. Dynamic Routing rules use Business Rules for routing the information.
- a. True
  - b. False
60. The Composite Instance title can be set using \_\_\_\_\_ functions.
- a. `setCompositeTitle()`
  - b. `setCompositeInstanceID()`
  - c. `setCompositeName()`
  - d. None of the above

- 61. BPEL code cannot be viewed from the console.
  - a. True
  - b. False
- 62. Cross Reference can be created from the console.
  - a. True
  - b. False
- 63. Security policies can be added to a composite at runtime in Oracle SOA Suite 11g.
  - a. True
  - b. False
- 64. Header properties in Mediator are automatically propagated to the target service.
  - a. True
  - b. False
- 65. Header properties in the inbound BPEL adapter are automatically propagated to target services.
  - a. True
  - b. False
- 66. Service Virtualization Endpoint Management can be achieved using Mediator.
  - a. True
  - b. False
- 67. Business Rules can be tested from JDeveloper.
  - a. True
  - b. False
- 68. Identify from the list the participant types used in Human Workflow.
  - a. Single approver
  - b. Parallel
  - c. Serial
  - d. FYI
  - e. All the above

69. Oracle FTP and File adapter do not support XA transaction.
- a. True
  - b. False
70. Identify which of the following adapters support XA transaction.
- a. JMS adapter
  - b. AQ adapter
  - c. Database adapter
  - d. All the above

# C

## Exam Reference

This section of the book will provide key points to be remembered for the exam and other additional details that are required to understand Oracle SOA Suite in detail.

### SOA concepts

The service-oriented architecture concepts are as follows:

- **Service Component Architecture (SCA)** is the key enabler of the 11g service platform.
- SDO provides a metadata API, which allows applications, tools, and frameworks to introspect the data model for a data graph.
- SDO specifications are different from SCA specifications. SCA promotes the use of SDO to represent the business data for assembly and transportation.
- Business events are not currently part of SCA specifications; they are in the process of getting incorporated into SCA specifications.
- JCA contributes towards the SOA application standard to establish communication between different applications or enterprise information system such as file, database, queue, topic, and so on.
- XQuery is an XML traversal path in an XML document to obtain information about a node or a subset of nodes.
- UDDI is a standard method for publishing and discovering SOA components.
- WS-ReliableMessaging supports standards to transmit messages with high reliability to enable the QOS. WS-Addressing, WS-I, WS-Security are standards that supplement the WS-RM standards.

- WS-Coordination supersedes WS-Transaction and WS-AtomicTransaction in defining the context management properties for Web Service transaction management.
- Oracle has intentionally not implemented nested SCA specifications as the nested composite application implementation leads to circular reference issue.

## SOA composite applications

The key components that build the SOA composite applications are as follows:

- Oracle BPEL process components used for process/service orchestration.
- Oracle Business Rule components used for centralized services/process-related business rule implementation and execution.
- Various adapter services such as file, JMS, and database adapters.
- Oracle Mediator components used for filter, data transformation, and routing.
- Oracle Human Task components used to define the human actions to be executed in the business process flow.
- EDN that supports an event-driven architecture for event-driven applications.
- Wires are used to create interaction between service end points, components, and references in SOA composites.
- A composite is the unit of deployment for SCA. It holds services that can be accessed remotely and it contains one or more components.
- SCA components are integrated together into one application and communicate with the outside world through binding components such as adapters and Web Services.
- The ADF BC component can be configured to publish events for the CRUD operations it can perform.
- SCA provides an assembly model for SOA-style applications.
- SCA components are basic elements of business functions in an SCA assembly. Components are combined into complete business solutions by SCA composites.
- **Oracle User Messaging Service (UMS)** provides a common service responsible for sending out messages from applications to devices.
- Service interactions within SOA Suite are handled using Oracle Mediator.

- Service Virtualization cannot be achieved using Mediator.
- Oracle Service Bus or Enterprise Service Bus is required to carry out Service Virtualization. Only mediation of a service can be achieved using Oracle Mediator.
- The BPEL/BPM Engine represents the orchestration component of SOA infrastructure.
- Oracle BPEL Engine supports BPEL 1.1 and 2.0.
- Incoming and outgoing XML documents for a Partner Link can be validated by setting the `validateXML` property on a Partner Link to `true`.
- **Metadata Service (MDS)** allows you to share common artifacts such as XML schemas across SOA composites. It supports two types of repositories, namely, file-based repository and database repository.
- The Service Infrastructure provides the internal message routing infrastructure capabilities for connecting components and enabling data flow.
- The Service Infrastructure is responsible for the internal routing of messages between service engines and binding components.
- Weblogic JMS is used as a messaging layer for delivering events through EDN. EDN can also be configured to use EDN-DB. EDN-DB uses an Oracle database as a backend store.
- Event definitions are stored in the EDL file.
- Event Delivery Network can work across SOA Suite.
- **Business Activity Monitoring (BAM)** is used for visualizing and monitoring business activity.
- Oracle Service Registry supports UDDI V3 standard.

## Working with adapters

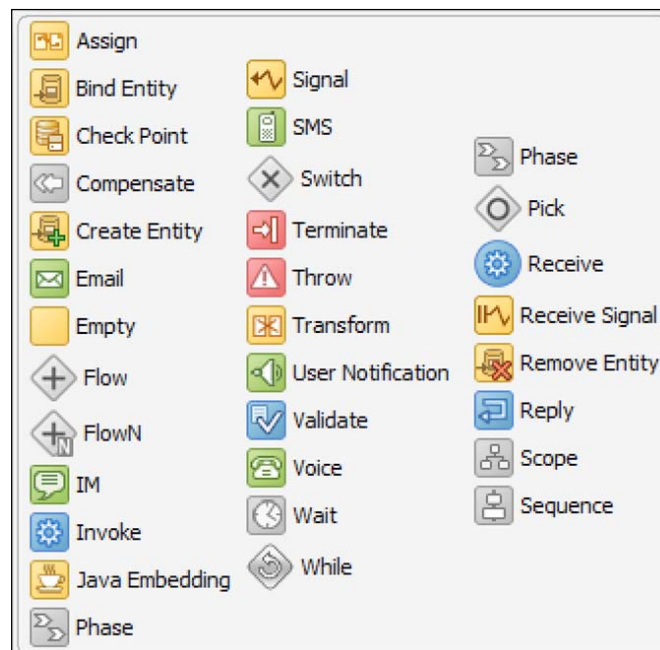
- Adapters are used in Oracle SOA Suite 11g to connect to non-service enabled backend systems.
- File adapter supports four types of operations: read file, write file, synchronous read, and list files.
- FTP adapter supports four types of operations: get, put, synchronous get file, and list files.
- Oracle FTP and file adapter do not support XA transaction.
- XA transaction is supported by all JCA-compliant adapters (for example: DB, JMS, AQ adapter).



- Inbound adapter-rejected messages are handled through fault policies.
- The SCA component should participate in the XA transaction and the other adapter should not have the idempotent property enabled.
- Fault policy doesn't work for both outbound adapters in XA mode and outbound adapters with Mediator in sequential routing.
- Oracle file adapter provides an option to delete the files after processing. This can be enabled by selecting delete after processing the checkbox in the **Adapter Configuration Wizard**.
- Larger payload support is available in Oracle File Adapter.
- The Oracle adapters support metadata service. The Oracle File Adapter and the FTP adapter provide the features to read the metadata information such as filename, file modified date, and so on. The 'Do not read file content' option in the file adapter helps to achieve this.
- Adapter Configuration Wizard is the required feature to generate the WSDL for both the request-response and event-notification service.

## Orchestrating services with BPEL

The following image shows the list of BPEL activities:



The standard BPEL constructs are given in the following table:

S.No	Construct	Details
1	Invoke	Provides the feature to invoke the required Partner Link.
2	Partner Link	Partner Link is a Web Service or adapter that is called from the BPEL process.
3	Receive	Receive activity can be used to receive the message to the BPEL process. This can be used at the start or in the middle of the BPEL process.
4	Reply	Reply activity is used to reply back to the caller of the message. This is used by the BPEL process in synchronous communication.
5	Assign	Assigns the variables to values or expressions.
6	Compensate	The Compensate activity is a business activity that compensates for the previous executed steps.
7	Empty	The Empty activity is used to represent that no action is required on the branch of Flow or While.
8	Terminate	The Terminate activity ends the BPEL process execution.
9	Throw	Throw activity throws the fault that is captured in the Catch block or the Fault that is constructed in the CatchAll block.
10	Wait	The Wait activity provides features to pause the BPEL process execution for a specified period of time.
11	Flow	The Flow activity is a unique activity for BPEL. It helps to run tasks in parallel. There are two types of Flow activities.
12	Pick	Pick is also a unique activity for BPEL. This activity is used to receive the message from various different Partner Links.  It is used in an asynchronous BPEL process. This can also be used at the start or in the middle of the BPEL process.
13	Scope	Scope is the activity used to group the set of BPEL activities for logically grouping a set of actions to handle faults, define local variables for manipulation, and so on.
14	Sequence	Sequence is another grouping construct that is used to sequence the list of BPEL activities.
15	Switch	Switch is used in a very similar way to Switch Case constructs in C programming language.
16	While	While is a looping construct that is used to execute the list of actions until the condition specified in While constructs return to false.
17	If (BPEL 2.0)	Switch is replaced with If in BPEL 2.0. This construct helps us perform the if-else operations.

S.No	Construct	Details
18	Repeat Until (BPEL 2.0)	This is an additional construct introduced in BPEL 2.0 to repeat certain actions until the condition is met.
19	ReThrow (BPEL 2.0)	This is an additional construct that allows you to re-throw the fault caught in the Fault Handler block.
20	ForEach (BPEL 2.0)	This is an additional construct introduced in BPEL 2.0. This will replace the FlowN construct by allowing parallel execution.

The Oracle extensions to BPEL constructs are given in the following table:

S.No	Construct	Details
1	E-mail	Activity to send an e-mail to any target recipient.
2	User Notification	Uses the UMS (Unified Messaging Framework) to send SMS, e-mail, IM, and voice messages.
3	Validate	Validates a variable data against the schema definition.
4	IM	Sends an instance message to the required person.
5	Checkpoint	Checkpoint activity helps to dehydrate the BPEL process to the database.
6	Phase	This is a special purpose activity which helps to construct an output message based on the business rules defined around the inputs message. This can be used as a replacement for data transformation where multiple condition checks are used.
7	Voice	This activity enables the sending of a telephone voice notification.
8	Signal	This activity is used in the master process to send a signal to the detailed process.
9	Receive Signal	This activity is used in the detail process to receive a signal sent from the master process.
10	Create Entity	This activity allows the creation of an entity variable.
11	Bind Entity	This activity allows the binding of an entity variable.
12	Transformation	This activity allows the calling of an XSLT function from BPEL.

- BPEL schema validation can be done to validate activity.
- BPEL exposed as a service can have multiple operations.
- The process will not be terminated when a fault is thrown from the BPEL process.
- Pick without Alarm can be the first activity in the BPEL process.
- Pick cannot be the very first activity in the BPEL process.
- Events can be generated from BPEL.

- Fault policies can be used to handle a composite fault as well as a component-level fault.
- `SyncMaxWait` is a BPEL container property that controls the synchronous response time of a BPEL. If the BPEL doesn't respond to the caller within this wait time, the BPEL transaction will be timed-out.
- The `doTransformation` function takes multiple input parameters for transformation.
- Wait, Checkpoint, Pick, Mid-process receive activities will dehydrate a BPEL process forcefully.

## Advanced BPEL concepts

The advanced BPEL concepts are as follows:

- BPEL provides rich features to handle faults.
- Faults can be managed through `Catch`, `CatchAll`, `Compensate`, and also through Fault policies.
- The purpose of the compensation activity is to do a business rollback rather than a transaction rollback of a physical transaction. `Compensate` is not used for an ACID purpose; it compensates for an asynchronous activity in the business process flow.
- The faults that are defined by WS-BPEL specifications are standard faults.
- A BPEL process in SOA Suite 11g supports the business and runtime fault categories.
- If a fault policy is defined, it overrides the fault handling in the BPEL process.
- Faults are notified and handled differently for synchronous and asynchronous invocations.
- The `bpelx:replay` fault is used to replay the scope by throwing a replay fault.
- Fault policy and fault policy bindings are defined in `fault-policies.xml` and `fault-bindings.xml`, and they are bound to a composite using `composite.xml`.
- A custom fault policy needs to implement the `oracle.integration.platform.faultpolicy.IFaultRecoveryJavaClass` interface.

- The order of precedence in a BPEL process, which the Fault policy execution follows is:
  - Fault policies specified in `composite.xml`
  - Fault handlers configured using the `Catch` block at scope level and the `CatchAll` at process level (main scope)
- BPEL supports both DOM and SDO data forms.
- The `FlowN` activity creates multiple flows equal to the value of `N`, which is defined at runtime based on the data available and logic within the process.

## Mediator

Mediator provides the following list of features:

- Routing between components in an SCA assembly.
- Validation of incoming messages into an SCA assembly.
- Transformation of data from one format to another within an SCA assembly.
- Filtering to allow the selection of components to invoke based on the message content.
- Oracle Mediator supports static and dynamic routing rules.
- XSL transforms can be used in Mediator to provide mappings between one data format and another.
- Validation of input data to Mediator can be done using the Schematron file. Input data can also be validated against an XSD using the `validate` syntax option in the Mediator option.
- Mediator has a feature to create events on target types for static or dynamic rules.
- Dynamic routing of messages is possible through the asynchronous interface in Mediator.
- Dynamic routing of rules in the Oracle Mediator component allows the externalizing of the routing logic to an Oracle Rules Dictionary.
- EDN uses JMS to deliver events from subscribers to publishers, but the configuration of JMS queues and topics and any associated filters is hidden from users of the EDN service.
- Business events are defined using the **Event Definition Language (EDL)** in Oracle SOA Suite 11g.
- Header properties are not propagated to the target service automatically. If the `pass through` property is enabled, it is propagated automatically.

- When the underlying WSDL is changed, the Mediator need not be rebuilt; it can be refreshed to obtain the changes in the WSDL.
- Switch case is the first action in Mediator, followed by transformation, assignment, and invoking of the required service/event.
- Oracle Mediator initiates a new transaction for processing each parallel run.
- The callout class must implement the `oracle.tip.mediator.common.api.IjavaCallout` interface.
- Callouts are available for both static and dynamic routings.
- Oracle Mediator provides support for setting rules based on message payload or message headers.
- Oracle Mediator provides support for synchronous and asynchronous request and response interactions.
- A routing rule can be either executed in parallel or sequentially.
- Oracle Mediator supports data transformation from one XML schema to another using XSL.
- Oracle Mediator provides support for validating the incoming message payload using a Schematron or an XSD file.
- Oracle Mediator supports subscribing to business events and raising business events.
- Oracle Mediator can process messages consisting of multiple parts.
- Oracle Mediator supports both manual error handling and error handling based on a fault policy.
- Oracle Mediator WSDLs can be modified by adding or deleting operations. After modifying the WSDL file, use the refresh WSDL dialog to synchronize the changes.
- The echo option is available for synchronous operations like request/reply and request/reply/fault.
- The echo option is not available for Oracle Mediator interfaces having request/reply/fault/callback WSDL files or for one-way WSDL files.
- The echo option is only available for synchronous operations when the routing rule is sequential.
- The priority property is applicable only to parallel routing rules.
- You cannot route an Oracle Mediator response to a two-way service. If you want to route a response to a two-way service, you should use a one-way Oracle Mediator between the first Oracle Mediator and the two-way service. The response should first be forwarded to the one-way Oracle Mediator, which in turn should call the two-way service.

- All the fault policies for a composite are loaded when the first error occurs. At runtime, Oracle Mediator checks whether there is any policy defined for the current error. If a fault policy is defined, then Oracle Mediator performs the action according to the configuration in the fault policies file. If there is no fault policy defined, then the default action of human intervention is performed.
- Error groups in Oracle Mediator are:  
`TYPE_DATA, TYPE_METADATA, TYPE_FATAL, TYPE_TRANSIENT, TYPE_INTERNAL.`
- Schematron files are not compiled or validated by the Oracle SCA compiler.

## Business Rules

The Business rules are as follows:

- Oracle Business Rule follows RETE Algorithm.
- Business rules can be used to make choices regarding routing rules. This can be done using the dynamic business rule option.
- Oracle Business Rule Engine supports four types of facts: Java facts, XML facts, RL facts, and ADF facts.
- Facts represent the data or business objects that can be used by rules.
- A global variable is similar to a public static variable in Java.
- Bucketsets define the data types of fact properties.
- Bucketsets are used as a constraint to define a set of acceptable values for a fact or a property.
- Links are used to link to a dictionary in the same application or in another application.
- Decision function is a function that is configured declaratively, without using RL language programming.
- A ruleset is a collection of one or more rules that is seen as a unit of execution.
- The Dictionary holds facts, rules, and rulesets.

## Human workflow

- The task routing service deals with the dispatching and assigning of tasks
- With Oracle ADF, you can design Oracle Human Task display forms that depict the human task in the SOA composite
- Human Workflow Engine contains the task management service, identity service, and notification service

## Monitoring and managing SOA 11g deployment

The monitoring and managing of SOA 11g deployment is done as follows:

- SOA Suite 11g provides the ability to simultaneously host multiple versions of a given composite
- In an enterprise manager, each instance is uniquely identified with an ECID
- Oracle Enterprise Manager SOA Suite 11g performs administrative tasks like configuring, monitoring, and managing of runtime artifacts
- An Oracle SOA Composite Application can be deployed using ANT scripts, enterprise manager, JDeveloper, or from command-line utilities
- The deployment unit of an SOA composite is the SCA archive file or a .sar file
- Business events could be monitored using enterprise manager of Fusion Middleware Console
- Shared data is deployed to MDS in **Metadata Archives (MAR files)**

## SOA 11g integration points

- Messages pass through the pipelines one-way; they cannot travel through both ways
- Proxy services have message flows and provide mediation capabilities such as data transformation
- Content-based routing is supported by the dynamic routing feature using XPath expressions
- Oracle service bus supports request, response, and error pipelines
- Oracle service Bus can monitor the performance of proxy and business services and fire alerts based upon SLA definitions



- Performance, service throttling, service pooling, service caching, and flexibility are services provided by OSB as enhancements to the ESB core architecture
- B2B supports HL7, RosettaNet, ebXML, and EDI, EDIFACT
- ODI is recommended for the large payload
- BAM server components consist of ActiveDataCache, EventEngine, and ReportCache
- BAM data will be stored in memory in a component called Active Data Cache
- **Oracle Complex Event Processing (CEP)** is a complete solution for building applications to filter, correlate, and process events in real-time so that downstream applications, service-oriented architectures, and event-driven architectures are driven by true, real-time intelligence

## Miscellaneous

- JDeveloper provides a Web Service test tool through which the Web Service can be tested and a response can be obtained using the tool
- Automated upgrade of 10g design-time artifacts is supported for Oracle BPEL and Oracle ESB
- Managed server is an instance of an Oracle WebLogic server used to host customer-deployed applications
- JDeveloper is used as an Oracle SOA composite editor
- OWSM gateways are not supported in Oracle SOA Suite 11g
- The following table lists the difference between Oracle SOA Suite 10g and 11g:

Topic	10g	11g
Application server	Oracle Application Server	Web logic
Service Bus	ESB	Mediator (Note: This will only mediate the request between the Oracle SOA components in the composite)
Orchestration engine	BPEL	BPEL – new activities like create, bind, remove entity, and signals
SDO support	Not available	Available in BPEL
Component architecture	Not available	SCA

Topic	10g	11g
Large file handling	Not available	Available
Scalable DOM support	Not available	Available
Logging support	Custom code	Through policy
DVM	Available in the ESB console	Can be created from JDeveloper
Cross reference	Available through the command-line tool	Can be created from JDeveloper
Security	OWSM is used	Security policies can be used
Business rules	Business rule console is used to update and edit it	Rules can be created and updated using JDeveloper
Events	Not available	The EDN framework available for publishing and subscribing to events
End-to-end tracking of application	Not available	ECID helps to track the instances end-to-end
Fault-policies	Only BPEL faults are supported	Mediator and BPEL faults can be handled
Streaming support in the adapter	Not available	Available
Runtime	Diversified	Unified
Transformation – multiple sources	Not available	Available
BAM	Sensor framework is used to integrate with BPEL	Sensor framework + BAM adapter are provided



# D

## Answers

The following are the answers and explanation to the questions found at the end of each chapter and appendix.

### Chapter 2

1. a: Creating a web service using WSDL is described to be a top-down approach in web service development. In this approach, WSDL is designed and used for creating the implementation logic. The top-down approach is supported by most of the SOA Suite vendors. Tools are available to generate Java, C, C#, and so on.
2. b: XQuery is not a subset of XPath. XQuery is an XML traversal path in an XML document for obtaining information about a node or a subset of nodes.



`bpelx:dehydrate` activity is provided as a replacement to `checkpoint` activity in the latest versions of Oracle SOA Suite.

3. b: UDDI is a standard method for publishing and discovering SOA components.
4. c: WS-ReliableMessaging supports standards to transmit messages with high reliability to enable the QOS. WS-Addressing, WS-I, and WS-Security are standards that supplement the WS-Reliable Messaging standards.
5. a: XML can be represented in binary form.
6. c: SCA provides a specification for SCA JCA binding rather than SCA JCA framework.
7. d: Groovy is a domain-specific language for Java. This language is not currently supported by SCA.

8. b: SDO specifications are different from SCA specifications. SCA promotes the use of SDO to represent the business data for assembly and transportation.
9. a: The property called auto-wire is provided at the composite level to auto-wire the components.
10. b: Business events are not currently part of SCA specifications; they are in the process of getting incorporated in SCA specifications.
11. a: The event definition for Oracle SOA Suite containers is defined using an XML-based language called **Event Definition Language (EDL)**. The EDL structure is followed through `edl.xsd`.
12. a: SCA specifications are extended to the description of consumers and producers of an event in `composite.xml`.

## Chapter 3

1. d: A business rule is a component used to define the business rule that will change the outcome or path of the business process flow.
2. d: File Adapter is a binding component while the other options specified are service components.
3. a: BPEL is used to orchestrate a business process.
4. c: Oracle Composite application is defined using `composite.xml`.
5. b: B2B Service Engine is a binding component to Oracle SOA Suite 11g.
6. b: SOA Fault Policy can be specified from the source view of the `composite.xml` file. The SOA Composite Editor cannot be used to configure Fault Policies.
7. b: Not all the components can be tested from design time. Only the business rule function can be tested from JDeveloper. We also have few limitations towards it.
8. a: Management, security, and optimization policies can be attached to Oracle SOA components.
9. a: DB Connection is required in order to create the MDS connection.
10. b: SCA is an open standard created for the assembly of components from different languages for building composite applications.

## Chapter 4

1. b: Adapter Configuration Wizard is the required feature to generate the WSDL for both request-response and event-notification service.
2. a: Oracle FTP and File Adapter do not support XA transactions.
3. d: All the adapters specified: Oracle JMS, AQ, and Database adapters support XA transaction.
4. b: Oracle adapters support Metadata Service. Oracle File Adapter and FTP Adapter provide features to read the metadata information, such as filename, file modified date, and so on. The **Do not read file content** option in the File Adapter helps to achieve this.
5. b: Inbound rejected messages are handled through fault policies.
6. a: Large payload support is available in the Oracle File adapter.
7. d: Batching and de-batching is supported in all adapters: Oracle FTP, File, and Database.
8. b: SCA components should participate in the XA transaction and another adapter should not have the **idempotent** property enabled.
9. c: Fault policy doesn't work for both outbound adapters in XA mode and outbound adapter in Mediator Sequential Routing.
10. a: The Oracle File adapter provides an option to delete the files after processing. This can be enabled by selecting the **Delete after Processing** checkbox in the Adapter Configuration Wizard.

## Chapter 5

1. b: The purpose of the compensation activity is to do a business rollback rather than a transaction rollback of a physical transaction. Compensate is not used for the ACID purpose; they compensate for an asynchronous activity in a business process flow.
2. b: Checkpoint activity is introduced in 11g. Checkpoint activity can be used to dehydrate a transaction in the middle of the business process execution.
3. a: Flow activity helps to execute the process in parallel.
4. b: Binding faults can be caught at the invoke scope level or the main scope level to catch any binding fault that is thrown from the process. It is essential to capture the system and binding faults. This can be done through fault policies as well.

5. a: This is a new function that was introduced in 11g Oracle SOA Suite to support multiple inputs to the XSL. This will help to aggregate the message structure using XSL.
6. b: Events can be generated using the Invoke activity from BPEL.
7. a: Pick without an alarm can be used to receive messages from multiple partner links.
8. b: `SyncMaxWait` is a BPEL container property that controls the synchronous response time of a BPEL. If the BPEL doesn't respond to the caller within this wait time, the BPEL transaction will be timed out.
9. a: It is possible to design BPEL with multiple operations exposed through the SCA service. This restriction was in place in Oracle BPEL Process Manager 10g, which requires a pick-based receive operation to select the right message to a process.
10. a: BPEL with a reply option is a synchronous BPEL process.
11. a: BPEL propagates a fault to the caller using the Reply activity.
12. b: A BPEL process doesn't get terminated when faults are thrown from the BPEL process. The process gets faulted; to terminate the process, the Terminate activity should be used, which is one of the dehydration activities.
13. b: Features to execute the Java code from the BPEL Engine are the extension features provided by Oracle. This is not natively supported by BPEL OASIS standards.
14. b: The synchronous BPEL process uses only one port to communicate with the caller.
15. d: The icon shown in this question represents checkpoint or dehydration.

## Chapter 6

1. a: Multiple `CatchAll` blocks can be used in a BPEL.
2. b: Multiple `CatchAll` blocks cannot be used per BPEL scope. Only one `CatchAll` block is allowed per BPEL scope.
3. a: Fault management framework is the right option. `Catch` and `CatchAll` blocks can be used only with BPEL processes.
4. a: Custom Java actions are available to handle the faults through the fault management framework.
5. b: At the least, fault binding definitions are required to utilize the fault management framework.

6. a: Fault management policies take higher precedence over `Catch` and `CatchAll` faults. Fault policies can be defined to re-throw the faults back to the `Catch` activities.
7. b: We can have multiple different fault policies defined per SOA installation and they can be bound to required composites.
8. a: Fault management framework doesn't differentiate between a sync and an async process. As a best practice, it is advisable to have a retry option for a sync process.
9. a: Fault management framework doesn't differentiate between a sync and an async process. As a best practice, it is advisable to have a retry option for an async process.
10. b: Fault policies can be used to handle both custom as well as system-defined faults.
11. a: Multiple compensation handlers can be defined per BPEL through multiple scopes. However, this cannot be true per BPEL scope.
12. b: Multiple compensation handlers can be defined per BPEL through multiple scopes. However, this cannot be true per BPEL scope.
13. b: The property alias required for a message correlation ID is defined in the WSDL file.
14. a: Correlation sets are defined in the BPEL file.
15. b: `getFaultAsString()` can be used only in the `CatchAll` block. It cannot be used in the `Catch` block, and it doesn't return any results.

## Chapter 7

1. a: Validation of input data to the Mediator can be done using a Schematron file. Input data can also be validated against an XSD using the **validate syntax** option in the Mediator.
2. b: Virtualization Endpoint Management cannot be achieved using the Mediator. Oracle Service Bus or Enterprise Service Bus is required to do the Virtualization Endpoint Management. Only mediation of a service can be achieved using the Oracle Mediator.
3. a: Mediator has a feature to create events on target types for static or dynamic rules.
4. a: Business Rules can be used to make a choice for routing rules. This can be done using the **dynamic business rule** option.
5. c: Dynamic routing of messages is possible only through the asynchronous interface in Mediator.



6. a: From Oracle SOA Suite 11g release, faults are handled through fault policies.
7. d: All the above patterns can support the echo option. The echo option is not supported as a part of the one-way async service.
8. b: Header properties are not propagated to target service automatically. If the pass through property is enabled, it is propagated automatically.
9. b: Switch case is the first action in Mediator, followed by transformation, assignment, and invoking the required service/event.
10. a: A Mediator can have multiple operations. For each operation, there can be one default routing rule.
11. b: When the underlying WSDL is changed, the Mediator need not be rebuilt; it can be refreshed to obtain the changes in the WSDL.
12. b: A Mediator initiated through events doesn't have the capability to echo the message.

## Chapter 8

1. a, b, and c: A task can be assigned to a user, group, or application role by statically or dynamically using XPath or based on rules.
2. a, b, c, and d: Reminders, escalations, renewal, and expirations are the different types of deadlines that can be associated with the tasks.
3. d: All of the above are correct.
4. a: Using Business Rules to route is the advanced option used in Oracle SOA Suite 11g.
5. a: Task routing can be achieved using the delegation rules option in the BPM Worklist Application.
6. b, c, and d: Configuring the Human Workflow component in Oracle SOA Suite 11g involves identity, security, and task routing services.
7. c and d: The Oracle BPM Worklist Application can be used to define delegation rules and task management.
8. c, d, and e: The Oracle BPM Worklist Application provides features to do task management, defining workflow delegation rules, and personalizing the task flow.
9. e: The Oracle BPM Worklist Application provides all the list of reports.

10. d: The Oracle BPM Worklist Application provides all the list of reports.
11. a: User-related information can be exported from one environment to another.
12. a, b, c, d, e, and f: The Oracle Human Workflow service provides all the list services in the options.
13. f: Oracle BPM Worklist Application provides all the statuses provided in the list.
14. a: Oracle Business Rules can be used to alter the flow of the Human Workflow.
15. a: The Oracle Human Workflow can be exposed as a service.
16. a: The Oracle Human Workflow can be called from the Mediator.
17. a: Approve and reject are the default outcomes that are specified in the Oracle Human Workflow.

## Chapter 9

1. b: Oracle Business Rules is the rule-based system that is a data-driven forward chaining system.
2. b: It is not safe to edit the rules by multiple users on the same dictionary.
3. a: A rule dictionary is externalized from the Business Process Execution layer and it can stand alone and change without affecting the other layers of the application.
4. d: Oracle Business Rules provide Rule Engine, Working Memory, and Inference Engine.
5. c: Oracle Business Rules can be accessed from Java and SOA through the decision functions.
6. a: Oracle Business Rules contain rule dictionary.
7. a: Rules are data or business objects on which the Rules Engine evaluates rule conditions.
8. d: All the examples stated in the questions apply to Business Rules.
9. a: Oracle Business Rules information is stored in a \*.rules file.
10. b: Bucketsets are used to define the list of values and list of ranges to be used in the Business Rule dictionary.

## Chapter 10

1. a: Oracle WebLogic provides support to secure web services.
2. b: Oracle Web Service Manager is integrated with Oracle WebLogic as a single entity in Oracle Fusion Middleware 11g.
3. e: Oracle Web Service Manager is composed of a Policy Manager, a WSM Agent, Meta Data Services, and Policy Inceptors.
4. e: Oracle Fusion Middleware 11g supports WS-ReliableMessaging, WS-Addressing, WS-Security, and MTOM-based policies.
5. a: WS-Policies can be attached from JDeveloper and Fusion Middleware Console.
6. h: All of the above.
7. e: All of the above.
8. f: All of the above.
9. c: Oracle Fusion Middleware control provides an option to attach policies and test it without redeploying the composites from JDeveloper.
10. b: Oracle Fusion Middleware control provides an option to attach policies and test it without redeploying the composites from JDeveloper.

## Chapter 11

1. a: Virtualization is the core theme of any ESB. Oracle Service Bus provides this feature to complete the Oracle SOA Suite and to complete the Oracle SOA stack.
2. a and d: JMS and RMI/EJB are the supported transport protocols.
3. b: Mediator cannot be used as a substitute for OSB. Mediator can be used only to interconnect the SOA components.
4. b: JDeveloper cannot be used to develop OSB components.
5. a, b, and c: OSB supports fan-out, dynamic routing, and split-join design patterns.
6. c: Oracle B2B supports dynamic routing of messages based on the trading partner and document type.
7. d: OSB supports transformation of data using XSL, XQuery, and MFL.
8. a: Oracle B2B is an EDI Gateway product in the Oracle SOA Suite of products.
9. a, b, and c: Oracle B2B and Oracle SOA can be integrated using the B2B Adapter or it can be integrated using AQ, JMS through AQ, and the JMS Adapter.

- 10. a: True is the correct answer.
- 11. a: Oracle has provided an ant task to deploy the agreements to target environments. The following command will help to deploy the Oracle artifacts to SOA runtime:  

```
ant -f ant-b2b-util.xml b2bdeploy
```
- 12. b: Oracle provides the feature to resubmit the business as well as application messages.
- 13. a: Oracle B2B 11g provides an option to auto-create agreements.
- 14. a: Oracle B2B 11g provides an option to store the partner contact information.
- 15. b: Oracle B2B is not part of Oracle EDN Suite.
- 16. b: Oracle B2B is registered as bindings in the Oracle SOA infrastructure.
- 17. a: Oracle B2B provides the features to auto-generate the functional acknowledgements.
- 18. b: Inbound Oracle B2B errors can be handled by reading the messages in IP\_IN\_QUEUE with b2berroruser as the consumer name.
- 19. a and b: IP\_IN\_QUEUE and IP\_OUT\_QUEUE are the queues provided by the Oracle B2B product.
- 20. b: Default queues provided by Oracle B2B is of JMS type.
- 21. a: Oracle B2B provides features to batch the messages.
- 22. a: Oracle B2B provides the feature to call Java programs and XSLTs through the callout feature.
- 23. b: Oracle B2B does not provide the feature to encrypt or decrypt the messages; these need to be implemented using the callout feature.
- 24. b: Oracle B2B does not provide the support for ebMS 2.0 specifications for error handling.
- 25. a and b: Complex Event Processing (CEP) is supported through Event Processing Language (EPL) and Continuous Query Language (CQL).
- 26. a and c: Oracle CEP is supported through EPL and CQL.
- 27. a and b: Mediator and BPEL can subscribe and publish events, but Human Workflow components can only publish events.
- 28. b: Fault handling is supported by the EDN layer.
- 29. b: Oracle CEP programs cannot be developed using JDeveloper. Eclipse is used to develop Oracle CEP programs.
- 30. a: Oracle CEP Visualizer is preinstalled along with the CEP server.

31. a, c, and d: Activity, fault, and variable are the different types of sensors that are supported by BPEL.
32. b, c and d: Oracle BAM server consists of Active Data Cache, Event Engine, and Report Cache.
33. a, b, c, and d: Oracle BAM server can be integrated with Oracle SOA using the BAM Adapter, BAM Sensor, JMS Adapter, and DB Adapter.
34. d: BAM Data Control allows ADF developers to build applications with a dynamic user interface that changes based on real-time business events.
35. a: Oracle BAM objects can be migrated using the ICommand utility.

## Chapter 12

1. a: The Oracle SOA Suite EM console provides the feature to deploy the SCA composites.
2. a: The Oracle SOA Suite EM console provides the feature to export the SCA composite to files.
3. a: The Oracle SOA Suite EM console provides the feature to deploy the SCA composites through the user authentication procedure. Only the users with the right deployment privileges can deploy the composites.
4. d: `ant-sca-deploy.xml` is used to deploy the composite to SOA runtime.
5. b: Deployment plans can be attached from the console to apply customization to SAR files.
6. c: Service Archive is the correct answer.
7. b: Cross references cannot be created from the SOA EM console. They can only be edited from the console.
8. b: Domain-value maps are deployed through SOA archive files and can be exported as files. They can be referenced in SCA runtime using the MDS URLs.
9. b: Business Rules can be edited from Oracle SOA Composer.
10. a: Deployment of an SOA composite can be controlled by a specific user using the roles option.
11. b: B2B is a binding in the Oracle SOA Suite architecture. It is not a Service Engine. Mediator, BPEL, Human Task, and Business Rules components are executed by the specific Service Engines.
12. b: BPEL code can be viewed from the console.
13. b: Shutting down the composite affects the existing running instances, as it does not allow the new instance to be created.

- 14. a: The Shutdown and Retire options do not allow the creation of new instances.
- 15. a: Only exposed services can be tested using the composite. Components that are present in the SCA Assembly but not exposed through the service interface cannot be tested.
- 16. a: Overriding the composite deployment changes the instance from state to stale.
- 17. b: Composites can be undeployed from the console using the undeploy option.
- 18. b: Changing composite properties does not require additional authentication.
- 19. b: The MBean Browser can be used to change the composite properties.
- 20. a: Policies can be attached to SOA composites at runtime.
- 21. a: The Oracle SOA Enterprise Manager console provides the feature to view rejected and faulted messages.
- 22. a: The Oracle SOA Enterprise Manager console provides the features to monitor the events generated through the Event Layer.

## Appendix A

- 1. a: BPEL expands to Business Process Execution Language.
- 2. b: Abstract WSDL need not contain a SOAP address.
- 3. b: XQuery is supported in Oracle BPEL through `ora:processXQuery()` custom XPath function.
- 4. b: XSL is used for the transformation of messages in BPEL.
- 5. a: OSR is used as a service registry to register the services for discovery using UDDI protocols.
- 6. a: Executable BPEL is an XML file.
- 7. a: Abstract BPEL is also an XML file.
- 8. b: Most of the Oracle technology adapters are JCA-compliant. Few of the application adapters are still in custom configuration and connection mode.
- 9. c: `composite.xml` stores SCA definition information.
- 10. a and b: Java code can be executed from Mediator as well as from BPEL.
- 11. c: The refresh button in the Mediator configuration screen can be used to change the `.mpplan` information.
- 12. a: Security and logging policies can be configured/added to a service from JDeveloper.
- 13. d: All of the above are correct.

14. a: `<terminate/>` is the correct answer.
15. e: All the above are correct.
16. b: False is the correct answer. `while` is a BPEL construct.
17. a: Transformation is the Oracle BPEL construct that uses the assignment construct from the BPEL construct definition.
18. a, b and c: Replay, validate, and e-mail are Oracle extension constructs.
19. b: BPEL, Mediator, and Human Workflow cannot be unit tested from JDeveloper. XSL and Business Rules can be unit tested from JDeveloper. However, unit test suites can be created and attached to the composite to access it from the console.
20. a: True is the correct answer.
21. a: Business Rules are used to configure the business policies.
22. b: Business rules developed from JDeveloper can be edited.
23. b: Composites can be undeployed from consoles directly.
24. b: Only the **Retire** option will affect the running instances.
25. a, b, c, and d: `Catch`, `CatchAll`, `Throw`, and `Compensate` are supported to handle faults in the BPEL process.
26. a: SOA Composer is used to edit the DVM and cross references.
27. a: Event definition information is stored in EDL files.
28. b: `Catch` block provides an option to catch a specific fault.
29. b: SAR is the extension for SOA Composites.
30. b: B2B is a binding engine in Oracle SOA Infrastructure.

## Appendix B

1. d: SCA supports implementation using Java, C++, and PHP, but support for Groovy is still not available.
2. c: SCA provides a specification for SCA JCA Binding rather than SCA JCA Framework.
3. b: Only asynchronous Mediator faults can be handled by fault policies.
4. a: Multiple input parameters can be used in XSL for transformation of messages in BPEL.
5. a: If you are using DB-based MDS, database connection is required to connect to MDS.

6. b: BPEL Dehydration status can be changed by using persistent policy parameters.
7. b: XSLs can be referenced from MDS.
8. e: All standards specified enable SOA in Oracle SOA Suite 11g.
9. e: All the above are correct.
10. b: SOA fault policies need to be specified using the `composite.xml` file.
11. b: Service Infrastructure helps to build the internal message transportation between different service engine components.
12. b: Analyzing existing applications and assets and building services around them is the bottom-up approach. Starting with the business process view and then drilling down to the service level is the top-down approach.
13. e: All the above are correct.
14. a and b: BPEL and Mediator support Java Callout.
15. b: The Refresh button in Mediator can be used to change the Mediator definition.
16. a: The WSDL snippet represents a synchronous BPEL.
17. d: All the above are correct.
18. a: `<terminate/>` is the correct answer.
19. e: All the above are correct.
20. b: `Pick` existed since BPEL 1.1 specifications.
21. e: Validation of an XML message received by the SOA-Infra layer can be enabled using a **Payload Validation** option available in common properties of SOA-Infra.
22. a: True is the correct answer.
23. e: All the above are correct.
24. b: The synchronous process can only wait until `SyncMaxWaitTime` is specified in SOA-Infra.
25. b: False is the correct answer.
26. a: `bpel.config.transaction` can take a value of `required` or `requiresNew` to specify the transactionality features of service components.
27. b: `SyncMaxWaitTime` is not applicable to BPEL in all scenarios. It only applies to the synchronous process invocations when the process has a dehydration point. If no breakpoint is identified, the entire process is executed by the client thread and returns a reply message.



- 28. b: It is not mandatory to have the `createInstance` attribute set to `true` for Mid-Process receive, and it is not possible to compile a process with the `createInstance` flag turned on for two receive activities.
- 29. b: Shutting down the composite will not terminate the running instance.
- 30. a, b, c, and d: Fault Management can be achieved using `Catch`, `CatchAll`, `Throw`, and `Compensate`.
- 31. a: SOA Composer is used to edit or change the DVM/Cross References.
- 32. a: Multiple endpoints can be specified to support fail over.
- 33. b: `Catch` activity can be used to catch the binding faults.
- 34. b: Composite definition is packaged as an SAR file.
- 35. a: XSL can be tested from JDeveloper.
- 36. b: Oracle BPEL Process Manager's implementation of the BPEL 2.0 specification does not support the IMA feature provided by BPEL.
- 37. b: Dynamic partner link support is not available in BPEL 2.0.
- 38. e: All the above are correct.
- 39. a: `for-each`, a BPEL construct can support the parallel execution of scope. `<forEach parallel="yes" counterName="ForEach1Counter" name="ForEach1">`
- 40. e: All the above are correct.
- 41. b and c: It can be used only in the `Catch` and `CatchAll` blocks of the Main scope.
- 42. b: The `skipCondition` attribute is not supported by BPEL 2.0.
- 43. a: True is the correct answer.
- 44. b: Fault policies can be referred from MDS.
- 45. b: `getFaultAsString` works with the `CatchAll` block.
- 46. a: True is the correct answer.
- 47. b: False is the correct answer.
- 48. a: True is the correct answer.
- 49. d: All the above are correct.
- 50. a: The in-flight BPEL process will not persist the messages from the transaction into the dehydration store.
- 51. d: All the above are correct.
- 52. a: True is the correct answer.
- 53. a: True is the correct answer.

- 54. c: XSL, XQuery, and MFL are used by OSB for data transformation.
- 55. d: All the above are correct.
- 56. c: Split-join helps to achieve the parallel execution of the services.
- 57. d: All the above are correct.
- 58. a and b: Business policies can be coded as if-then rules or decision tables.
- 59. a: True is the correct answer.
- 60. d: `setCompositeInstanceTitle()` is an advanced Xpath function that helps to set the title of the instance.
- 61. b: BPEL source code can be viewed from the console.
- 62. b: Cross references can be viewed and edited only from the SOA Composer.
- 63. a: True is the correct answer.
- 64. b: False is the correct answer.
- 65. b: False is the correct answer.
- 66. b: Service Virtualization can be done by the Mediator; but it cannot accomplish the Virtualization Endpoint Management.
- 67. a: True is the correct answer.
- 68. e: All the above is the correct answer.
- 69. a: True is the correct answer.
- 70. d: All of the above are correct answers.



# Index

## Symbols

1Z0-451 Oracle SOA Foundation  
Practitioner exam 9  
360Commerce 13  
<Created> timestamp 217  
<Nonce> element 217

## A

Access level section, Human Workflow  
definition 165  
access rules 158  
Active Studio 233  
Active Viewer 233  
Adapter Configuration Wizard Flow 69  
adapter design-time configuration  
about 69  
Direct Binding 87  
file adapter 70-74  
FTP adapter 76-78  
HTTP binding wizard 88, 89  
MQ series adapter 80-83  
socket adapter 84-86  
adapters  
about 42, 59, 313  
categories 42, 43, 59  
runtime configuration 90  
working with 313  
ADF-BC Adapter Service 67  
ADF BC component 312  
ADF-BC Facts 185  
ADF-BC Service 43  
ad-hoc routing 156  
Administrator 233  
AdminServer 13

advanced BPEL concepts  
about 317  
compensation handling 128  
correlation 129  
exception handling 118  
fault management framework 124, 125  
fault policy resolution process 127  
Advanced Visual Technology 13  
Agile 12  
Amber Point 11, 40  
AppForge 12  
application design context 21  
Application Explorer 61  
application integration context 21  
application-level security  
about 211, 212  
Web Services Addressing (WS-Addressing)  
218  
Web Services Security (WS-Security) 213  
WS-Policy 216  
WS-ReliableMessaging (WS-RM) 218  
WS-SecurityPolicy 216, 217  
WS-Security tokens 213  
XML encryption (data confidentiality) 212  
XML signature 213  
applications adapters  
about 68  
framework architecture 68  
Application view, SOA Composite Editor  
46, 47  
AQ Adapter  
about 43, 66  
dequeue features 66  
enqueue features 66  
features 67

- Architect** 233
- Assign construct** 105, 315
- Assignment section, Human Workflow**
  - definition 160, 162
- asymmetric cipher model** 210
- asynchronous bi-directional BPEL process**
  - about 102-104
  - features 102-104
- asynchronous - one way BPEL process**
  - about 101
  - features 101
- ATG** 12

## B

- B2B** 44
- B2B Adapter** 43, 67
- BAM**
  - about 14, 43, 313
  - integrating, with Oracle SOA Suite 233
- BAM Adapter** 43, 67
- BEA** 11
- Bharosa** 11
- big bang model** 24
- bilateral authentication** 211
- BPEL**
  - about 14, 15, 23, 30, 31, 40, 158
  - advanced concepts 117, 317
  - communication techniques 99-104
  - components activities 105, 106
  - conditional branching 108-110
  - human task, calling from 172
  - integrating, with Java 110-112
  - parallel flow 108, 110
  - partner links 105, 106
  - services, orchestrating 314-317
- BPEL4WS** 25
- BPEL activities** 314-317
- BPEL constructs**
  - about 315
  - Assign 315
  - Compensate 315
  - Empty 315
  - Flow 315
  - ForEach (BPEL 2.0) 316
  - If (BPEL 2.0) 315

- Invoke 315
- Oracle extensions 106, 107
- Partner Link 315
- Pick 315
- Receive 315
- Repeat Until (BPEL 2.0) 316
- Reply 315
- ReThrow (BPEL 2.0) 316
- Scope 315
- Sequence 315
- Switch 315
- Terminate 315
- Throw 315
- Wait 315
- While 315

### BPEL Engine Properties

- managing 282

### BPEL Engine Statistics 278

### BPEL process

- Business Rule, including as activity 200
- Business Rules, integrating 196
- human task, invoking from 167-171

### bpel:exec methods 112

### BPM 23

### BPMN Engine Properties

- managing 283

### Bridgestream 11

### BSE 68

### bucketsets 185, 320

### Business Activity Monitoring. *See* BAM

### business event 30

### Business Process Execution Language. *See* BPEL

### Business Process Management. *See* BPM

### business process orchestration context 22, 40

### Business Rule component

- wiring, through SOA Composite 197-199

### Business Rule implementation

- style, selecting 193

### Business Rules

- about 30, 41, 177, 178, 320
- components 179
- concepts 182-186
- editing, from SOA composer 204
- examples 41, 177
- features 178

- including, as activity to BPEL process 200
- integrating, with BPEL process 196
- testing, from JDeveloper 201, 202
- Business Rules, concepts**
  - bucketsets 185
  - decision function 187
  - decision tables 186
  - facts 185
  - rule action 184
  - rule condition 184
  - rule dictionary 186
  - rule language 186
  - rules 183
  - rule set 184
- Business Service Engine.** *See* BSE
- business significance, SOA** 22, 23
- C**
  - C# 23
  - C++ 31
  - callbacks 158
  - Captivation** 11
  - Catch activity** 118
  - CatchAll activity** 118
  - CEP** 322
  - certificate authorities (CA)** 214
  - certification objectives, Oracle SOA**
    - about 15
    - adapters 16
    - advanced BPEL concepts 16
    - human workflow 16
    - monitoring and managing SOA 11g
      - deployment 17
    - Oracle Business Rules Concepts 17
    - orchestrating services, with BPEL 16
    - secure services and composite applications 17
    - SOA 11g Integration Points 17
    - SOA Composite Applications 15
    - SOA concepts 15
    - working, with mediator components 16
  - certification, Oracle SOA**
    - need for 18
  - ClearApp** 11
  - communication techniques, BPEL**
    - asynchronous - bi-directional 102-104
    - asynchronous - one way 101
    - synchronous 99
  - Compensate activity** 105, 315
  - compensation** 128
  - compensation handler** 128
  - Complex Event Processing.** *See* CEP
  - componentization** 31
  - ComponentPalette, SOA Composite Editor** 45
  - components, Business Rules**
    - Rule Author 179
    - Rule Engine 181
    - Rule Language 180
    - Rule SDK 180
  - components, Oracle Human Workflow architecture**
    - Human Workflow service engine 150, 151
    - Human Workflow services 150-152
    - Oracle Worklist Application 150-152
  - components, Oracle SOA Suite 11g**
    - about 40
    - adapters 42
    - B2B 44
    - BPEL 40
    - Business Activity Monitoring (BAM) 43
    - Business Rules 41
    - Human Workflow 42
    - MDS 44
    - Mediator 41
  - Composite Application.** *See* SOA
    - Composite Application**
  - composite application context** 22
  - composite applications**
    - exception handling 118
  - composites**
    - about 312
    - Jdeveloper, used for deploying 247-251
    - undeploying 259
  - composite security**
    - and web services 210
  - concepts, Oracle Human Workflow**
    - access rules 158
    - ad-hoc routing 156
    - callbacks 158
    - events 158
    - notifications 157
    - participant 155

- routing policy 156
- stages 158
- standalone human task 158
- task assignment 155
- task deadlines 157
- task forms 157
- task stakeholders 156
- concepts, SOA 311, 312**
- conditional branching, BPEL 108, 110**
- configuration details, MQ series adapter 80-83**
- configuration plan**
  - creating, for SCA composite 260-262
- configuring**
  - file adapter 70-74
  - FTP adapter 76-78
  - Mediator 140, 141
  - Mediator service 138, 139
- Conformia Software 12**
- Context Media 11**
- Convergin 12**
- correlation 102, 129**
- correlation, concepts**
  - property 129
  - property alias 129
- CRUD operations 312**
- custom faults 119**

**D**

- Database Adapter 43, 66**
- data confidentiality 211**
- DataScaler 11**
- Data section, Human Workflow definition 160**
- Deadline section, Human Workflow definition 163**
- decision function 320**
- Decision Function 187**
- Decision Table 186**
- Demantra 12**
- deploy agreements, Oracle B2B 237**
- deploying**
  - SOA composite application 247
- deployment configuration plan**
  - generating, SOA Composite Editor used 48
- Direct Binding 87**

- Direct Binding Adapter 67**
- Distinguished Name (DN) 214**
- document protocols, Oracle B2B**
  - custom document definition 235
  - cXML 235
  - EDIEL 235
  - EDIFACT 235
  - HL7 235
  - NCDCP 235
  - OAG 235
  - UBL 235
  - X12 235
  - xCBL 235
- doTransformation function 317**

## E

- ECID 14**
- EDA**
  - about 29
  - overview 29
  - real world example 29
  - with Oracle SOA Suite 11g 30
- EDN 15**
- EDN-DB 313**
- EJB 31**
- EJB Service 43**
- EJB Service Adapter 67**
- Empty activity 105, 315**
- Engine-level Dashboard view 277**
- Enterprise Application Integration 59**
- Enterprise Manager (EM)**
  - about 15, 246
  - launching 245
  - used, for managing SOA composite application 264-266
- Enterprise Manager Grid Control, SOA Management Packs**
  - features 287, 288
- Enterprise Service Bus. *See* ESB**
- error groups, Mediator**
  - about 144
  - TYPE\_DATA 144
  - TYPE\_FATAL 144
  - TYPE\_INTERNAL 145
  - TYPE\_METADATA 144
  - TYPE\_TRANSIENT 145

- ESB 23
- eServGlobal's Universal Service Platform 12
- e-Test 11
- Event Definition Language (EDL) 318
- Event Delivery Network (EDN) 16, 30
- Event Driven Architecture. *See* EDA
- event manager 29
- events
  - about 29, 158
  - used, for configuring Mediator 142
- events, Human Workflow definition 167
- evidence store service 152
- exam details, Oracle SOA certification 18
- exception handling
  - about 118
  - activities 118
  - in composite applications 118
- exception handling, activities
  - catch 118
  - CatchAll 118
  - reply 118
  - terminate 118
  - throw 118
- exceptions 118
- export features, SOA EM console 275
- export options, SOA composite application 275, 276
- Extensible Markup Language. *See* XML
- Extensible Stylesheet Language Transformations. *See* XSLT

**F**

- facts 185
- Fault Chooser 121
- fault management framework 124, 125
- fault management, Mediator 143
- Fault Policy Framework 143
- fault policy resolution process 127
- features, asynchronous bi-directional BPEL process 102, 104
- features, asynchronous - one way BPEL process 101
- features, Business Rules 178
- features, Fusion Middleware Control 264
- features, Human Workflow 42

- features, Mediator
  - about 137, 318, 320
  - Consistent Fault Policy Framework with BPEL 137
  - Content-Based and Header-Based Routing 137
  - Dynamic Routing 137
  - EDN support 137
  - Java callout 137
  - Mediator Echo Support 137
  - Multiple Part Message Support 137
  - Schematron Validation 137
  - Sequential and Parallel Routing of Messages 137
  - Synchronous and Asynchronous Interactions 137
  - Transformations 137
  - XPath-based Assignment 137
- features, Oracle B2B
  - agreement management 234
  - certificate management 234
  - document management 234
  - integration support 234
  - trading partner management 234
- features, Oracle Human Workflow
  - identity management 154
  - management rules 154
  - notifications 154
  - report and audit trails 154
  - task assignment 154
  - task forms 154
  - task routing 154
  - worklist application 154
- features, Oracle Human Workflow architecture 149, 150
- features, Oracle Service Bus (OSB)
  - data transformation 231
  - dynamic routing 231
  - location transparency 231
  - message enrichment 231
  - patterns 231
  - protocol transformation (adapters) 231
  - service orchestration 231
  - service pooling 231
- features, Oracle SOA Suite 11g
  - about 15
  - BPEL 15



- EDN 15
- Enterprise Manager 15
- mediator 15
- features, SOA Composite Editor 45**
- features, synchronous BPEL process 99**
- File Adapter**
  - about 43, 65, 313
  - configuring 70-74
- Flow activity 106, 108, 315**
- FlowN activity 108**
- forEach activity 109**
- ForEach (BPEL 2.0) construct 106, 316**
- ForEach construct 108**
- FTP adapter**
  - about 43, 65, 313
  - configuring 76-78
- functions, JCA 60**
- Fusion Middleware Control**
  - features 264
- Fusion Middleware Security architecture 218**

## G

- General section, Human Workflow definition 160**

- G-Log 12
- GoldenGate 11
- granularity 24

## H

- Haley 12
- high-level Oracle Human Workflow architecture 151**
- homepage, Oracle B2B 11g 236**
- HotSip 12
- HTTP Binding Adapter 67**
- HTTP binding wizard 88, 89**
- human services**
  - designing 158, 159
- human task**
  - calling, from BPEL 172
  - designing 158, 159
  - invoking, from BPEL process 167-171
- Human Workflow**
  - about 30, 42, 321
  - features 42

## Human Workflow definition

- sections 159

## Human Workflow definition, sections

- Access level 165
- Assignment 160, 162
- Data 160
- Deadlines 163
- events 167
- General 160
- Notification 164

## Human Workflow service engine 150, 151

## Human Workflow services

- about 150, 151
- evidence store service 152
- identity service 152
- notification service 152
- runtime config service 152
- task assignment service 151
- task metadata service 151
- task query service 152
- task report service 152
- task service 151
- user metadata service 151

## Hyperion 12

## HyperRoll 11

## I

## IBM 40

## identity service 152

## If (BPEL 2.0) construct 106, 315

## i-flex 12

## If-then rule

- example 193

## industry standards, Oracle B2B

- CIDX 235
- ebXML 235
- EDI 235
- PIDX 235
- positional flat files 235
- RosettaNet 235
- UBL 235
- UCCNet 235

## Innobase 11

## integration points, SOA 11g 321

## Interlace Systems 12

## Internet Engineering Task Force (IETF) 211

interoperability 23  
Invoke construct 105, 315  
iOpen standards 9  
iWay Software 40

## J

J2EE Connector Architecture. *See* JCA  
J2EE standards 60  
Java  
    about 11, 23, 31  
    BPEL, integrating with 110-112  
Java Facts 185  
Java Key Store (JKS) 234  
Java Message Service (JMS) 24  
Java virtual machine (JVM) 22  
JCA  
    about 60, 311  
    functions 60  
JD Edwards 68  
JDeveloper  
    about 14, 61, 322  
    Business Rules, testing from 201, 202  
    SAR file, generating 251-253  
    used, for deploying composites 247-251  
JDeveloper Rule Designer  
    used, for creating rules 188-195  
JMS Adapter 43, 66  
JMX API 286  
JSR 322 60

## K

Kerberos  
    about 214  
    principal 214  
    realm 214  
Key Distribution Center (KDC) 214  
key performance indicators (KPIs) 43, 232

## L

landing screen, Oracle Worklist Application 153  
legacy adapters 42  
links 320  
LODESTAR 13

LogicalApps 12  
loose coupling 22, 24

## M

management policies 220  
MAR files 321  
Market2Lead 12  
MBean Browser 286  
MDS 44, 219, 313  
MDS Repository 286  
Mediator  
    about 30, 41, 135, 158  
    configuring 140, 141  
    configuring, for subscribing to event 142  
    error groups 144, 145  
    exam objectives 136  
    example 41  
    fault management 143  
    features 136-320  
    overview 135  
    routing rules, creating 138, 139  
Mediator Engine Properties  
    managing 283  
Mediator service  
    configuring 138, 139  
    creating 138, 139  
message formats, Oracle B2B  
    MIME 235  
    SMIME 235  
    SOAP 235  
    XML 235  
    XMLDSig 235  
    XMLEncrypt 235  
message pipelines 231  
Message Transmission Optimization Mechanism. *See* MTOM  
Metadata Archives. *See* MAR files  
Meta Data Store. *See* MDS  
MetaSolv Software 12  
MFL 231  
modes, SSL  
    bilateral authentication 211  
    no authentication 211  
    one-way authentication 211  
Moniforce 11  
MQ Adapter 43, 66

**MQ series adapter**  
configuration details 80-83  
**MTOM** 25, 220  
**mValent** 11

## **N**

**Net4Call** 12  
**Netsure Telecom Limited** 12  
**nonce** 217  
**Notification section, Human Workflow**  
definition 164  
notification service 152

## **O**

**OASIS** 31, 40  
**Oblix** 11  
**OctetString** 11  
**Oracle**  
about 9, 40  
design-time environment, for adapters 62  
products 10-13  
services 10-13  
**Oracle Adapter Framework** 61, 62  
**Oracle Advanced Queuing (AQ)** 24  
**Oracle Applications** 43  
**Oracle Applications Adapter** 67  
**Oracle B2B**  
about 234  
deploy agreements 237  
document protocols 235  
features 234  
industry standard protocols 235  
message formats 235  
metrics report 237  
reference integration architecture 238  
transport protocols and exchanges 235  
**Oracle B2B 11g**  
enhanced features 235  
homepage 236  
**Oracle B2B metrics report** 237  
**Oracle B2B reference integration**  
architecture 238  
**Oracle BAM ADF control** 233  
**Oracle BAM architecture** 232

**Oracle BAM server** 233  
**Oracle BAM web applications**  
about 233  
Active Studio 233  
Active Viewer 233  
Administrator 233  
Architect 233  
**Oracle Business Rule Architecture** 179  
**Oracle Business Rules** 156  
**Oracle E-Business Suite** 149  
**Oracle Enterprise Manager Fusion**  
Middleware Control 219  
**Oracle extensions, for BPEL constructs**  
Bind Entity 107, 316  
Checkpoint 106, 316  
Create Entity 107, 316  
E-mail 316  
Email 106  
IM 106, 316  
Java Embedding 107  
Phase 107, 316  
Receive Signal 107, 316  
Signal 107, 316  
Transformation 107, 316  
User Notification 106, 316  
Validate 106, 316  
Voice 107, 316  
**Oracle Fusion Middleware 11g Release 1**  
(11.1.1), policies  
management policies 220  
Message Transmission Optimization  
Mechanism (MTOM) 220  
security policies 220  
WS-Addressing 220  
WS-ReliableMessaging 219  
**Oracle Fusion Middleware Database** 219  
**Oracle Human Workflow**  
about 149  
concepts 155  
features 154  
overview 149  
**Oracle Human Workflow architecture**  
components 150  
features 149, 150  
**Oracle Justification**  
URL, for certification 18

- Oracle Service Bus (OSB)**
  - about 30, 230
  - features 231, 232
- Oracle SOA certification**
  - exam details 18
  - need for 18
  - objectives 15-17
  - registration procedure, for exams 19
  - URL, for registration procedure 19
- Oracle SOA Consultant/Developer community 9**
- Oracle SOA Suite**
  - about 23, 229
  - answers, for self review questions 325-339
  - BAM, integrating with 233
  - exam reference 311-323
  - post assessment test 299-309
  - pre-assessment test 293-297
- Oracle SOA Suite 10g**
  - versus Oracle SOA Suite 11g 13, 14, 322, 323
- Oracle SOA Suite 11g**
  - about 13, 59
  - components 40-43
  - features 15
  - versus Oracle SOA Suite 10g 13, 14, 322, 323
- Oracle SOA Suite 11g, EDA**
  - about 30
  - process, for publishing events 30
  - process, for subscribing events 30
- Oracle SOA Suite 11g, EDN**
  - features 30
- Oracle SOA Suite 11g, SCA 34**
- Oracle Studio 61**
- Oracle technology adapters 42**
- Oracle Worklist Application**
  - about 150, 152
  - landing screen 153
  - URL, for accessing 153
- Oracle WSM Agent 219**
- Oracle WSM Component**
  - Metadata Store (MDS) 219
  - Oracle Enterprise Manager Fusion Middleware Control 219
  - Oracle Fusion Middleware Database 219
  - Oracle WSM Agent 219
  - Oracle WSM Policy Manager 219
  - Policy Interceptors 219
- Oracle WSM Policy Manager 219**
- Orchestration Engine 14**
- OSB patterns**
  - split-join 232
  - VETRO 231
- OSB product architecture 230**
- OWSM 224**
- OWSM agents 224, 225**
- OWSM gateways 322**

**P**

- packaged-application adapters 42**
- parallel flow, BPEL 108, 110**
- participant 155**
- Partner Link 105, 315**
- Passlogix 11**
- payment adjustment request scenario 150**
- PeopleSoft 11, 12, 68, 149**
- Phase Forward 12**
- PHP 31**
- Pick activity 106, 315**
- POJO 31**
- policies 177, 219**
- Policy Interceptor Pipeline 224**
- Policy Interceptors 219**
- Portal Software 12**
- post assessment test, Oracle SOA Suite 299-309**
- pre-assessment test, Oracle SOA Suite 293-297**
- Primavera 12**
- principal 214**
- product group, Oracle**
  - about 11
  - applications 12
  - database 11
  - industry solutions 12, 13
  - middleware 11
  - server and storage 12
- ProfitLogic 13**
- Project Explorer view, SOA Composite Editor 46, 47**
- property 129**
- property alias 129**

**Public Key Infrastructure (PKI)** 210

**publishing events**

process 30

## **R**

**realm** 214

**real world example, EDA** 29

**Receive activity** 105, 315

**registration procedure, Oracle SOA**  
certification 19

**Relsys** 12

**Repeat Until (BPEL 2.0) construct** 106, 316

**Reply activity** 105, 118, 315

**ResourcePalette, SOA Composite Editor** 46

**RETE algorithm** 181, 320

**Retek** 13

**ReThrow (BPEL 2.0) construct** 106, 316

**reusability** 22, 24

**RL Facts** 185

**rollback fault** 119

**routing policy** 156

**routing rules, Mediator**

creating 138, 139

**Rule Action** 184

**Rule Author** 179

**Rule Condition** 184

**Rule Dictionary** 186

**Rule Engine**

about 179-181

Interface Engine 181

Rule-Base 181

Working Memory 181

**Rule Engine working model** 181

**Rule Language** 179, 180, 186

**rules**

about 177, 183

creating, in JDeveloper Rule Designer 188-195

**Rule SDK** 179, 180

**ruleset** 320

**Rule Set** 184

**runtime config service** 152

**runtime configuration, for adapters** 90

## **S**

**SAML**

about 211-214

federation 215

identity propagation 215

proprietary web single sign-on (SSO) 215

web browser cookies, limitations 215

web services security 214

**SAML assertions** 215

**SAML framework**

assertions 215

bindings 215

profiles 215

protocols 215

**Sample Business Rule Decision service**

description 55

**Sample Component type description** 55

**sample human task**

creating, steps 168-171

**sample JCA file** 56

**SAP** 40, 68

**SAR file**

deploying, from SOA EM console 254-258

deploying, SOA EM SOA deployment

wizard used 251

generating, through JDeveloper 251-253

**SCA**

about 311

reviewing 31

with Oracle SOA Suite 11g 34

**SCA composite**

configuration plan, creating for 260-262

**SCA (Service Component Architecture)** 13

**Scope activity** 106, 315

**SDO** 311

**Secerno** 11

**Secure Socket Layer.** *See* SSL

**Security Assertion Markup Language.** *See*

SAML

**security policies** 220

**Sequence construct** 106, 315

**server authentication** 211

**service** 31

## **service adapters**

- ADF-BC Adapter Service 67
- AQ Adapter 66
- B2B Adapter 67
- BAM Adapter 67
- Database Adapter 66
- Direct Binding Adapter 67
- EJB Service Adapter 67
- File Adapter 65
- FTP Adapter 65
- HTTP Binding Adapter 67
- JMS Adapter 66
- MQ Adapter 66
- Oracle Applications Adapter 67
- Socket Adapter 67
- Third Party Adapter 68
- Web Service Adapter 68

## **service bus 230**

**Service Component Architecture.** *See* SCA  
**service configuration, Socket Adapter 84,**  
85, 86

## **service data object (SDO) 67**

## **service design, SOA**

- about 23
- granularity 24
- interoperability 23
- loose coupling 24
- resuability 24

## **Service Engine properties**

- managing 281-287

## **service management, SOA**

- about 24
- versioning, of services 24

## **service orientation**

- benefits 22

**Service Oriented Architecture.** *See* SOA

**Service-Oriented Architecture.** *See* SOA

## **services**

- orchestrating, with BPEL 314-317

## **service virtualization 231**

**Siebel 12, 68, 149**

**Sigma Dynamics 11**

**Silver Creek Systems 12**

**Simple Object Access Protocol.** *See* SOAP

**Skywire Software 13**

**Sleepycat 11**

## **SOA**

- about 9, 21
- application design context 21
- application integration context 21
- business process orchestration context 22
- business significance 22, 23
- composite application context 22
- concepts 21, 23, 311, 312
- new technologies, learning 23
- service design 23
- service management 24
- standards, for enabling 24, 25

## **SOA 11g**

- integration points 321

## **SOA 11g deployment**

- managing 321
- monitoring 321

## **SOA Application Policy configuration**

- example 209

## **SOA Composer**

- about 179
- Business Rules, editing from 204

## **SOA Composite**

- Business Rule component, wiring through 197-199

## **SOA Composite Application**

- about 39, 209, 312, 313
- creating, steps 48-56
- deploying 247
- export options 275, 276
- managing 270-274
- managing, Enterprise Manager used 264-266
- monitoring 270-274
- undeploying 247

## **SOA Composite Editor**

- about 44
- Application view 46, 47
- ComponentPalette 45
- deployment configuration plan, generating 48
- features 45
- Project Explorer view 46, 47
- ResourcePalette 46
- WS Policies, configuring 47

## **SOA configuration plan 260-262**

- SOA EM console**
  - export features 275
  - SAR file, deploying from 254-258
- SOA EM SOA deployment wizard**
  - used, for deploying SAR file generated through JDeveloper 251
- SOA farms 268**
- soa-infra**
  - managing 281-287
- SOA Infrastructure Common Properties**
  - managing 282
- SOA Management Packs**
  - about 245
  - Enterprise Manager Grid Control 287, 288
- SOAP 25**
- SOAP Envelope Header element 212**
- SOA performance**
  - about 279
  - monitoring 279, 280
- SOA Service Engines**
  - BPEL Engine Statistics 278
  - Engine-level Dashboard view 277
  - monitoring 276
  - properties, managing 281-287
- SOA, standards**
  - about 25
  - BPEL4WS 25
  - MTOM 25
  - SOAP 25
  - UDDI 25
  - WS-Coordination 25
  - WSDL 25
  - WS-ReliableMessaging 25
  - WS-Security 25
  - XML 25
  - XOP 25
  - XSD 25
  - XSLT 25
- SOA Systems 40**
- SOA topology 268**
- Socket Adapter**
  - about 43, 67
  - service configuration 84-86
- Sophoi 12**
- SPL WorldGroup 13**
- Spring 31**
- SSL**
  - about 210, 211
  - modes 211
- stages 158**
- standalone human task 158**
- standard BPEL constructs**
  - about 105
  - Assign 105
  - Compensate 105
  - Empty 105
  - Flow 106
  - ForEach (BPEL 2.0) 106
  - If (BPEL 2.0) 106
  - Invoke 105
  - Partner Link 105
  - Pick 106
  - Receive 105
  - Repeat Until (BPEL 2.0) 106
  - Reply 105
  - ReThrow (BPEL 2.0) 106
  - Scope 106
  - Sequence 106
  - Switch 106
  - Terminate 105
  - Throw 105
  - Wait 105
  - While 106
- Stellent 11**
- subscribing events**
  - process 30
- Sun 12**
- Sunopsis 11**
- Switch construct 315**
- synchronous BPEL process**
  - features 99
- SyncMAXwait property 317**
- system faults**
  - about 119
  - bindingFault 119
  - remoteFault 119
- T**
  - Tacit Software 11**
  - Tangosol 11**
  - task assignment 155**

- task assignment service 151
- task deadlines 157
- task forms 157
- task metadata service 151
- task query service 152
- task report service 152
- task service 151
- task stakeholders 156
- technology adapters 61
- Telephony@Work 12
- TempoSoft 12
- Terminate activity 105, 118, 315
- Third Party Adapter 68
- Thor Technologies 11
- Throw activity 105, 118, 315
- TimesTen 11
- Transport Layer Security (TLS) 211
- transport protocols and exchange patterns, Oracle B2B
  - AQ 235
  - AS1 235
  - AS2 235
  - ebMS 235
  - file 235
  - FTP 235
  - HTTP 235
  - HTTPS 235
  - JMS 235
  - MLLP 235
  - RNIF 235
  - SFTP 235
  - SMTP 235
  - TCP/IP 235
- TripleHop 11

## U

- UDDI 25, 311
- UDDI registry 216
- undeploying
  - SOA composite application 247
- Universal Description, Discovery, and Integration. *See* UDDI
- User Messaging Service (UMS) 312
- user metadata service 151
- username token 213

## V

- Validate-Enrich-Transform-Route-Operate pattern. *See* VETRO pattern
- VETRO pattern 231
- Virtual Iron 12

## W

- Wait activity 105, 315
- Web Service Adapter 68
- Web Services
  - about 43, 210
  - and composite security 210
- Web Services Addressing (WS-Addressing) 218
- Web Services Description Language. *See* WSDL
- Web Services Security (WS-Security) 213
- web service standards, SOA
  - WS-Business Process standards 28
  - WS-Management standards 28
  - WS-Messaging standards 26
  - WS-Reliability standards 27
  - WS-Security standards 27
  - WS-Service Description and Discovery standards 26
  - WS-Transaction standards 27
- Web Service Test Tool 322
- While construct 106, 315
- workflow 149
- WS-Addressing policies 220
- WS-AtomicTransaction 312
- WS-Business Process standards 28
- WS-Coordination 25, 312
- WSDL 23, 25, 119
- WS-Management standards 28
- WS-Messaging standards 26
- WS Policies
  - configuring, with SOA Composite Editor 47
- WS-Policy
  - about 216
  - configuration steps 221
- WS-Reliability standards 27
- WS-ReliableMessaging (WS-RM) 25, 218, 219, 311



- WS-Security** 25
- WS-SecurityPolicy**
  - about 216
  - structure 217
- WS-Security standards** 27
- WS-Security tokens**
  - about 213
  - Kerberos ticket 214
  - SAML token 214, 215
  - username 213
  - X.509 certificate 214
- WS-Service Description and Discovery**
  - standards 26
- WS-Transaction** 312
- WS-Transaction standards** 27

## **X**

- X.509 certificate**
  - example 213
- X.509 digital certificate** 214
- XML** 25
- XML-binary Optimized Packaging.** *See* XOP
- XML encryption** 212
- XML Schema Definition.** *See* XSD
- XML signature** 213
- XOP** 25
- XQuery** 231, 311
- XSD** 25
- XSL** 231
- XSLT** 25



**Thank you for buying  
Oracle SOA Infrastructure Implementation  
Certification Handbook (1Z0-451)**

## **About Packt Publishing**

Packt, pronounced 'packed', published its first book "Mastering phpMyAdmin for Effective MySQL Management" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: [www.packtpub.com](http://www.packtpub.com).

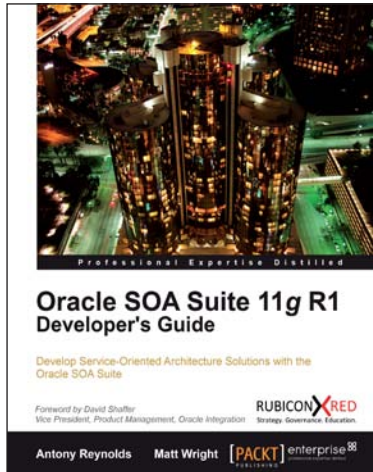
## **About Packt Enterprise**

In 2010, Packt launched two new brands, Packt Enterprise and Packt Open Source, in order to continue its focus on specialization. This book is part of the Packt Enterprise brand, home to books published on enterprise software – software created by major vendors, including (but not limited to) IBM, Microsoft and Oracle, often for use in other corporations. Its titles will offer information relevant to a range of users of this software, including administrators, developers, architects, and end users.

## **Writing for Packt**

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to [author@packtpub.com](mailto:author@packtpub.com). If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.



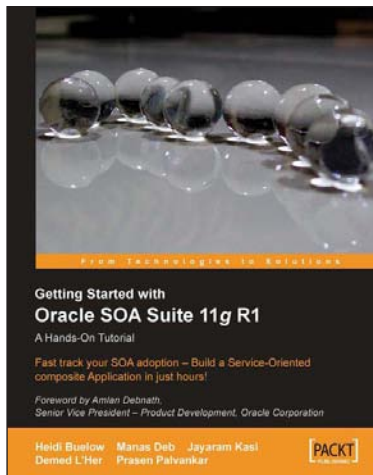
## Oracle SOA Suite 11g R1 Developer's Guide

ISBN: 978-1-84968-018-9

Paperback: 720 pages

Develop Service-Oriented Architecture Solutions with the Oracle SOA Suite

1. A hands-on, best-practice guide to using and applying the Oracle SOA Suite in the delivery of real-world SOA applications
2. Detailed coverage of the Oracle Service Bus, BPEL PM, Rules, Human Workflow, Event Delivery Network, and Business Activity Monitoring
3. Master the best way to use and combine each of these different components in the implementation of a SOA solution



## Getting Started With Oracle SOA Suite 11g R1 – A Hands-On Tutorial

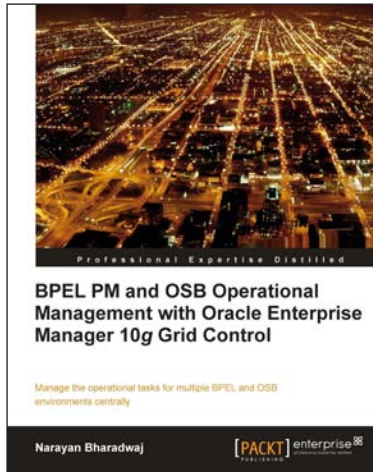
ISBN: 978-1-847199-78-2

Paperback: 482 pages

Fast track your SOA adoption – Build a Service-Oriented composite Application in just hours!

1. Offers an accelerated learning path for the much anticipated Oracle SOA Suite 11g release
2. Beginning with a discussion of the evolution of SOA, this book sets the stage for your SOA learning experience
3. Includes a comprehensive overview of the Oracle SOA Suite 11g Product Architecture

Please check [www.PacktPub.com](http://www.PacktPub.com) for information on our titles



## **BPEL PM and OSB operational management with Oracle Enterprise Manager 10g Grid Control**

ISBN: 978-1-847197-74-0      Paperback: 248 pages

Manage the operational tasks for multiple BPEL and OSB environments centrally

1. Monitor and manage all components of your SOA environment from a central location
2. Save time and increase efficiency by automating all the day-to-day operational tasks associated with the SOA environment
3. Step-by-step exercises to set up the framework to effectively manage Oracle SOA products



## **Getting Started with Oracle BPM Suite 11gR1 – A Hands-On Tutorial**

ISBN: 978-1-84968-168-1      Paperback: 536 pages

Learn from the experts – teach yourself Oracle BPM Suite 11g with an accelerated hands-on learning path brought to you by Oracle BPM Suite Product Management team members

1. Offers an accelerated learning path for the much-anticipated Oracle BPM Suite 11g release
2. Set the stage for your BPM learning experience with a discussion into the evolution of BPM, and a comprehensive overview of the Oracle BPM Suite 11g Product Architecture
3. Discover BPMN 2.0 modeling, simulation, and implementation

Please check [www.PacktPub.com](http://www.PacktPub.com) for information on our titles